لا نهار وآلاف الليالي المظلمة

دورة حياة ثغرات يوم الصفر وبرمجيات إكسبلويت وأوقات ظهورها

ليليان أبلون (Lillian Ablon)، أندي بوغارت (Andy Bogart)

RAND CORPORATION

لا نهار وآلاف الليالي المظلمة

دورة حياة ثغرات يوم الصفر وبرمجيات إكسبلويت وأوقات ظهورها

ليليان أبلون (Lillian Ablon)، أندي بوغارت (Andy Bogart)

للحصول على مزيدٍ من المعلومات حول هذا المنشور، الرجاء زيارة الموقع الإلكتروني:
www.rand.org/t/RR1751

نشرته مؤسسة RAND ، سانتا مونيكا، كاليفورنيا
© حقوق الطبع والنشر لعام 2017 محفوظة لصالح مؤسسة RAND
RAND® علامة تجارية مسجلة.

صورة الغلاف: صورة مركبة لأيلين دلسون لا روسو
مستمدة من صور أجيل ليوناردو، وماتييمو، وبياكايا؛ بموافقة غيتي إماجيز

Composite image by Eileen Delson La Russo.
Adapted from images by Agil_Leonardo, Matejmo, and Byakkaya; courtesy of Getty Images.

حقوق الطبع والنشر الإلكتروني محدودة
هذه الوثيقة والعلامة (العلامات) التجارية الواردة فيها محمية بموجب القانون. يتوفر هذا التمثيل للملكية الفكرية الخاصة بمؤسسة RAND للاستخدام لأغراض غير تجارية حصريًا. يحظر النشر غير المصرَّح به لهذا المنشور عبر الإنترنت. يصرح بنسخ هذه الوثيقة للاستخدام الشخصي فقط، شريطة أن تظل مكتملة بدون إجراء أي تعديل عليها. يلزم الحصول على تصريح من مؤسسة RAND لإعادة إنتاج أو إعادة استخدام أي من الوثائق البحثية الخاصة بنا، بأي شكل كان، لأغراض تجارية. للمزيد من المعلومات حول تصاريح إعادة الطباعة والربط على المواقع الإلكترونية، الرجاء زيارة صفحة التصاريح في موقعنا الإلكتروني:
.www.rand.org/pubs/permissions

مؤسسة RAND مؤسسة بحثية تعدّ حلولًا لتحديات السياسات العامة للمساهمة في جعل المجتمعات من حول العالم أكثر أمانًا، سلامة، صحة وازدهارًا. تعَدّ مؤسسة RAND مؤسسة غير ربحية، حيادية وملتزمة بالصالح العام.

لا تعكس منشورات مؤسسة RAND بالضرورة آراء عملاء ورعاة الأبحاث الذين يتعاملون معها.

ادعم مؤسسة RAND
وتبرع بمساهمة خيرية معفاة من الضريبة على الموقع الإلكتروني التالي:
www.rand.org/giving/contribute

www.rand.org

تمهيد

يستمر النقاش السياسي حول ما إذا كان يتعين على الحكومة الأمريكية، أو أي حكومة أخرى، الاحتفاظ بما يسمى "ثغرات يوم الصفر" (zero-day software vulnerabilities) أو الكشف عنها لكي يصبح بالإمكان إصلاحها.[1] فالذين لديهم علم بثغرات يوم الصفر قد يعدّون "برمجيات إكسبلويت" (exploits)، أي مجموعة رموز تستغل الثغرة، من أجل ولوج أجزاء أخرى من النظام وتطوير رمز خاص بهم والتصرف باعتبارهم مديرين متحكمين أو إتمام بعض الخطوات الأخرى. لكن الكثيرين يخشون من أن إبقاء هذه الثغرات سرية قد يعرّض الأشخاص الذين يستخدمون هذه البرمجية الهشة لهجمات خبيثة ومحاولات أخرى لجمع معلوماتهم الشخصية. فضلًا عن ذلك، تترتب على الأمن الإلكتروني والمسؤولية التي قد تنتج عن الهجمات وعمليات القرصنة وعمليات اختراق البيانات باستخدام ثغرات يوم الصفر (zero-day vulnerabilities) تداعيات كبيرة بالنسبة إلى المستهلكين والشركات والجهات الضامنة الأمريكية وإلى النظام القضائي المدني على نحو أشمل.

وغالبًا ما يؤجج النقاش حول ما إذا كان يجب الاحتفاظ بهذه الثغرات أو الكشف عنها مدى التداخل المحتمل بين ثغرات يوم الصفر أو برمجيات إكسبلويت التي تحتفظ بها الحكومة الأمريكية وتلك التي يخزنها خصومها. إذا كان للطرفين المخزونات نفسها، يرى البعض أنه لا جدوى من إبقائها خاصة، حيث أن تداخلًا أصغر قد يبرّر الاحتفاظ بها. لكن في ظل غياب المعلومات حول التداخل أو أي مؤشرات ملموسة تستند إلى بيانات فعلية، يصعب اتخاذ قرار مستنير بشأن التخزين.

ومن أجل معالجة هذه المسألة، كان لمؤسسة RAND نفاذ نادر إلى مجموعة من المعلومات الغنية حول ثغرات يوم الصفر وبرمجيات إكسبلويت. نستعرض في هذا التقرير مجموعة البيانات باستخدام تطبيقات جديدة لأساليب إحصائية تقليدية من أجل الكشف عن معلومات معمّقة حول القطاع وإعداد بعض المؤشرات الأولية المرتبطة بحالة وديمومة ومعدلات اصطدام ثغرات يوم الصفر وبرمجيات إكسبلويت. كما نتطرق إلى وقت العمل الضروري لإنشاء برمجية إكسبلويت. وتوفر نتائج هذا البحث نتائج مستخلصة من بيانات حقيقية لثغرات يوم الصفر وبرمجيات إكسبلويت من شأنها دعم الأمثلة البديلة التقليدية وآراء الخبراء، واستكمال الجهود الحالية لإعداد إطار عمل يُعنى باتخاذ قرار حول ما إذا كان يجب الكشف عن ذاكرة ثغرات يوم الصفر

[1] ثغرات يوم الصفر هي ثغرات لم يتم نشر أي طريقة لإصلاحها أو تعديلها بشكل علني. ويشير مصطلح ثغرة يوم الصفر إلى عدد الأيام التي علم خلالها بائع البرمجية بوجود ثغرة (ليبيكي Libicki، أبلون Ablon وويب Webb، 2015) . تُعتبر ثغرات يوم الصفر وبرمجيات أكسبلويت مفيدة في العمليات الإلكترونية – سواء تلك التي يشنها المجرمون أو الجيوش أو الحكومات – وكذلك في المجالات الدفاعية (مثلًا اختبار الانتشار) والأكاديمية.

وبرمجيات إكسبلويت أو الاحتفاظ بها، وتوفير المعلومات اللازمة لنقاشات السياسة في ما يخص التخزين والكشف عن الثغرات.

وقد يكون هذا البحث قيمًا بالنسبة إلى مجموعة واسعة من أصحاب الشأن، وبخاصة صناع السياسات الذين يتخذون قرارات بشأن كيفية الحدّ من هشاشة الدولة مع الاحتفاظ بخيارات صلبة للعمليات الإلكترونية.

تمّ تمويل هذا المشروع بفضل مساهمات خيرية من داعمي RAND وأعضاء مجلس المراقبين التابع لمعهد RAND للعدالة المدنية وجهات داعمة أخرى لمؤسسة RAND، فضلًا عن دخل العمليات.

معهد RAND للعدالة المدنية

يُعنى معهد RAND للعدالة المدنية بتحسين النظام القضائي المدني من خلال تزويد صناع السياسة والرأي العام ببحث دقيق وحيادي. تحدد دراساته الاتجاهات السائدة في الدعاوى القضائية وتعرّف بخيارات السياسة في ما يخص المسؤولية والتعويض والأنظمة وإدارة المخاطر والتأمين. ينطلق المعهد من تقليد قديم لأبحاث مؤسسة RAND يتسّم بمقاربة متعددة الاختصاصات وتجريبية إزاء قضايا السياسة العامة والمعايير الدقيقة للجودة والموضوعية والاستقلالية.

تحظى أبحاث معهد العدالة المدنية بدعم هبات مجموعة من المصادر تشمل مؤسسات واتحادات تجارية ومهنية وأفراد ووكالات حكومية ومؤسسات خاصة. وتخضع جميع تقاريرها لمراجعة نظيرة ويتمّ توزيعها على نطاق واسع في أوساط صناع السياسة وممارسي القانون والأعمال وباحثين آخرين والرأي العام.

يشكّل معهد العدالة المدنية جزءًا من معهد RAND للعدل والبنية التحتية والبيئة، وهو قسم تابع لمؤسسة RAND يُعنى بتحسين السياسة واتخاذ القرارات في مجموعة واسعة من مجالات السياسة، تشمل العدالة المدنية والجنائية وحماية البنية التحتية والأمن القومي وسياسات النقل والطاقة وكذلك سياسات البيئة والموارد الطبيعية. للمزيد من المعلومات حول معهد RAND للعدالة المدنية، يرجى زيارة موقع www.rand.org/icj أو التواصل مع المدير على العنوان icjdirector@rand.org.

نرحّب بأسئلتكم وتعليقاتكم التي يمكنكم توجيهها إلى الكاتبة الرئيسية ليليان أبلون (Lillian Ablon) (Lillian_Ablon@rand.org). للمزيد من المعلومات عن معهد RAND للعدالة المدنية، يمكنكم زيارة الموقع التالي: www.rand.org/icj أو التواصل مع المدير على العنوان icjdirector@rand.org.

المحتويات

تمهيد iii
الأشكال والجداول vii
ملخص ix
شكر وعرفان xvii

الفصل الأول
مقدمة 1
معلومات قليلة متوافرة حول مدى انتشار ثغرات يوم الصفر أو استعمالها أو فوائدها أو أضرارها 1
هل يجدر بالحكومة الأمريكية الكشف عن ثغرات يوم الصفر؟ 4
ثمة العديد من الاعتبارات التي يرغب أصحاب الشأن بمعالجتها 5
المسائل البحثية والغرض من هذا البحث 6
الجمهور المستهدف من هذا البحث 7
تجزئة مجال ثغرة يوم الصفر 7
البيانات المستخدمة في هذا البحث 11
منهجية البحث وجمع البيانات 13
تنظيم هذا التقرير 16

الفصل الثاني
نقاش إضافي حول ثغرات يوم الصفر 17
طبيعة ثغرات يوم الصفر 17
أسس تطوير برمجيات إكسبلويت والاعتبارات المتعلقة بها 18
دورة تطوير برمجيات إكسبلويت 20
الأشخاص في نطاق ثغرات يوم الصفر 20
نماذج الأعمال 22

الفصل الثالث
تحليل البيانات 27
1. الحالة: هل الثغرة هي ثغرة يوم صفر حقًا؟ هل هي "حية" (مجهولة للعامة) أم "ميتة" (معلومة للآخرين)؟ 28
2. الديمومة: إلى متى ستبقى الثغرة غير مكتشفة وغير معلنة 33

3. معدل الاصطدام: ما هو احتمال أن يكتشف الآخرون الثغرة ويفصحون عنها؟ 42
4. التكلفة: ما هي تكلفة تطوير برمجية إكسبلويت لاستغلال ثغرة؟ 46

الفصل الرابع
استنتاجات وتداعيات 51
الاستنتاج 1: قد يكون إعلان ثغرة حية (مجهولة للعامة) أو ميتة (معلومة للعامة) مضللًا أو مبسطًا للغاية 51
الاستنتاج 2: إن متوسط أمد الحياة المتوقع لبرمجيات إكسبلويت يبلغ 6.9 سنوات بعد اكتشافها الأولي، لكن حوالى 25% من برمجيات إكسبلويت لن تصمد لأكثر من عام ونصف في حين أن نسبة 25% أخرى ستصمد لأكثر من 9.5 سنوات 52
الاستنتاج 3: لم تشر خصائص ثغرة ما إلى حياة طويلة أو قصيرة، لكن قد تسعى تحليلات مستقبلية إلى مقارنة فيروس لينكس (Linux) مع أنواع أخرى من المنصات، ومراقبة أوجه الشبه بين رمز مصدر مفتوح وآخر مغلق وتجمعات مختلفة لنوع فئة برمجية إكسبلويت 54
الاستنتاج 4: اكتشفت أطراف أخرى نحو 5.7% من الثغرات لمجموعة محددة من ثغرات يوم الصفر بعد مرور سنة وأفصحت عنها 55
الاستنتاج 5: فور اكتشاف ثغرة يمكن استغلالها، يكون الوقت الضروري لتطوير برمجية إكسبلويت فعالة بالكامل سريعًا نسبيًا، حيث يبلغ متوسطه 22 يومًا 57
توصيات أخرى للدفاع 58
توصيات أخرى للهجوم 59
هل يجدر إيلاء ثغرات يوم الصفر هذا القدر من الاهتمام؟ 60
هل يجب تخزين الثغرات أم لا؟ 60
بعض التحذيرات بشأن بياناتنا 61
أبحاث المتابعة 63

الملحقات
الملحق A دورة تطوير برمجية إكسبلويت 65
الملحق B الباحثون عن الثغرات: من الذي يبحث عن الثغرات؟ 73
الملحق C كيف أثرت عمليات التخفيف على قابلية الاستغلال: دراسة حالة الاستغلال في بنية البيانات المكدسة مقابل المكونة 79
الملحق D اصطدامات قريبة 81
الملحق E شراء برمجية إكسبلويت لاستغلال ثغرة يوم صفر: بعض التكاليف والاعتبارات السعرية 83
الملحق F أشكال وجداول إضافية 89
الملحق G المزيد من المعلومات حول البيانات 97
الملحق H المسرد 103

المراجع 109

الأشكال والجداول

الأشكال

الشكل 1.1 الفهم المشترك لمخزون الثغرة 8
الشكل 1.2 نظرة مبسّطة على هوية من يشمله مخزون الثغرة 10
الشكل 1.3 نظرة مبسّطة على قرارات الاحتفاظ 10
الشكل 1.4 بياناتنا 12
الشكل 3.1 تجزئة بسيطة لحالة بياناتنا 30
الشكل 3.2 تجزئة إضافية لحالة بياناتنا 30
الشكل 3.3 الثغرات التي اكتشفها الباحثون عن الثغرات في BUSBY في كل عام 31
الشكل 3.4 الثغرات التي اكتشفها الباحثون عن الثغرات في BUSBY في كل عام 32
الشكل 3.5 تقديرات احتمال الصمود وفق كابلين –ماير 36
الشكل 3.6 منحنيات كابلين–ماير لنوع الثغرات والمنصة المتضررة ونوع رمز المصدر ونوع فئة برمجية إكسبلويت 38
الشكل 3.7 أنواع الثغرات مع مرور الوقت 44
الشكل 3.8 توقيت التداخل بين موعد اكتشاف جهتين مستقلتين لثغرة (2002-2016) 45
الشكل 3.9 تواتر فترة تطوير برمجية إكسبلويت 48
الشكل A.1 دورة حياة تطوير برمجية إكسبلويت 65
الشكل B.1 طول السير المهنية للباحثين عن الثغرات ضمن BUSBY 74
الشكل B.2 السنوات التي اكتشف خلالها الباحثون الثغرات ضمن BUSBY 75
الشكل B.3 موسمية باحثي الثغرات: الأشهر التي وجد خلالها باحثو BUSBY للثغرات ثغرات يمكن استغلالها 77
الشكل C.1 نوع فساد الذاكرة، العدد سنويًا 80
الشكل F.1 مقارنة نماذج بارامترية مع كابلين–ماير 91
الشكل F.2 وظيفة خطورة ميسرة 91
الشكل F.3 فاصل زمني لمدة 365 يومًا، مع النسب المئوية بحسب المجموعة والفاصل الزمني 96

الجداول

الجدول 3.1 وصف مختلف الحالات 31
الجدول 3.2 تقديرات أمد الحياة العامة ومن حيث خصائص برمجيات إكسبلويت المستقاة من نماذج الصمود الأسي غير المعدلة (n=127) 40

الجدول B.1 متوسط طول السيرة المهنية المسجلة لصائدي الأخطاء ومطوري برمجيات إكسبلويت في BUSBY 76
الجدول F.1 تواتر الخصائص على مستوى برمجيات إكسبلويت من بين 127 برمجية محددة 89
الجدول F.2 جدول الحياة، جميع البيانات 90
الجدول F.3 تقديرات معدل الخطورة من نماذج كوكس للخطورة النسبية، مع تعديل للخصائص أو بدونه كما هو مبين 94
الجدول F.4 أنواع الثغرات بحسب الفاصل الزمني، النسب 95
الجدول G.1 تعداد تواتر البيانات 97

ملخص

إن ثغرات يوم الصفر هي ثغرات في البرمجيات لم يتم نشر أي طريقة لإصلاحها أو تعديلها بشكل علني. ويشير مصطلح ثغرة يوم الصفر إلى عدد الأيام التي علم خلالها بائع البرمجية بوجود ثغرة (ليبيكي Libicki وأبلون Ablon وويب Webb، 2015). ويستخدم المهاجمون ثغرات يوم الصفر من أجل السعي وراء منظمات وأهداف تواظب على مواكبة برامج الإصلاح، أما الذين يهملون الأمر فيمكن أن يتعرضوا لهجوم عبر الثغرات التي لديها برامج إصلاح لكن لم يتمّ تطبيقها. وعليه، تكون ثغرات يوم الصفر وبرمجيات إكسبلويت مفيدة في العمليات الإلكترونية، سواء تلك التي يشنها المجرمون أو الجيوش أو الحكومات، وكذلك في المجالات الدفاعية (مثلًا اختبار الانتشار) والأكاديمية. وحتمًا، بدأت نماذج الأعمال والأسواق بالاستعداد للمداولة أو بيع ثغرات يوم الصفر القابلة للاستغلال.

يستمر النقاش السياسي حول ما إذا كان يتعين على الحكومة الأمريكية، أو أي حكومة أخرى، الاحتفاظ بما يسمى ثغرات يوم الصفر أو الكشف عنها لكي يصبح بالإمكان إصلاحها. فالذين لديهم علم بثغرات يوم الصفر قد يعدّون "برمجيات إكسبلويت"، أي مجموعة رموز تستغل الثغرة، من أجل ولوج أجزاء أخرى من النظام أو تطوير رمز خاص بهم أو التصرف باعتبارهم مديرين أو إتمام بعض الخطوات الأخرى. لكن الكثيرين يخشون من أن إبقاء هذه الثغرات سرية قد يعرّض الأشخاص الذين يستخدمون هذه البرمجية الهشة لهجمات خبيثة ومحاولات أخرى لجمع معلوماتهم الشخصية.

وغالبًا ما يؤجج النقاش حول ما إذا كان يجب الاحتفاظ بهذه الثغرات أو الكشف عنها مدى التداخل المحتمل بين ثغرات يوم الصفر أو برمجيات إكسبلويت التي تحتفظ بها الحكومة الأمريكية وتلك التي يخزنها خصومها. إذا كان للطرفين المخزونات نفسها، يرى البعض أنه لا جدوى من إبقائها خاصة، حيث أن تداخلًا أصغر قد يبرّر الاحتفاظ بها. لكن في ظل غياب المعلومات حول التداخل أو أي مؤشرات ملموسة تستند إلى بيانات فعلية، يصعب اتخاذ قرار مستنير بشأن التخزين.

ومن أجل معالجة هذه المسألة، كان لمؤسسة RAND نفاذ نادر إلى مجموعة من المعلومات حول ثغرات يوم الصفر وبرمجيات إكسبلويت. وإنها مجموعة بيانات غنية، إذ اكتشف آخرون وجود بعض برمجيات إكسبلويت هذه في حين لم يُكتشف البعض الآخر. وتغطي هذه المجموعة 14 عامًا (منذ 2002 وحتى 2016) وتحوي معلومات حول أكثر من 200 برمجية

إكسبلويت لاستغلال ثغرات يوم الصفر والثغرات التي استغلتها، حيث إن أكثر من نصفها غير معلوم.[1]

نستعرض في هذا التقرير مجموعة البيانات باستخدام تطبيقات جديدة لأساليب إحصائية تقليدية من أجل الكشف عن معلومات معمّقة حول القطاع وإعداد بعض المؤشرات الأولية المرتبطة بحالة ثغرات يوم الصفر وبرمجيات إكسبلويت وديمومتها، إضافةً إلى إمكانية اكتشافها من الآخرين (ما يُعرف بـ"معدل الاصطدام").[2] كما نتطرق إلى وقت العمل الضروري لإنشاء برمجية إكسبلويت لاستغلال ثغرة يوم الصفر. وتوفر نتائج هذا البحث نتائج مستخلصة من بيانات حقيقية لثغرات يوم الصفر وبرمجيات إكسبلويت التي من شأنها تعزيز الأمثلة البديلة التقليدية وآراء الخبراء، واستكمال الجهود الحالية لإعداد إطار عمل يُعنى باتخاذ قرار حول ما إذا كان يجب الاحتفاظ بذاكرة ثغرات يوم الصفر وبرمجيات إكسبلويت أو الكشف عنها، وتوفير المعلومات اللازمة لنقاشات السياسة القائمة في ما يخص التخزين والكشف عن الثغرات.

النتائج المستخلصة

أسفر بحثنا عن عدد من النتائج المستخلصة المثيرة للاهتمام. ونسلّط الضوء على اثنتين منها على سبيل المثال، ونفضّل الأخرى أدناه:

1. لبرمجيات إكسبلويت وثغراتها الكامنة متوسط أمد حياة متوقع طويل (6.9 سنوات)
2. بالنسبة إلى مجموعة محددة من ثغرات يوم الصفر، وبعد مرور سنة، اكتشف طرف خارجي نحو 5.7% منها

قد يكون التخزين مفيدًا لمن يركزون على الهجوم، ويفضل الباحثون عن الثغرات المتطورون من الناحية الفنية تخزين الثغرات التي اكتشفوها بدلًا من الكشف عنها. وسيكون المدافعون معرضين دومًا لثغرات يوم الصفر وسيرغبون على الأرجح في الكشف عن الثغرة عند اكتشافها وإصلاحها.

لم تشر بياناتنا إلى وجود أي ثغرات "أقوى" أو "أضعف" من غيرها من حيث مقاومة اكتشافها والكشف عنها. قد يكون تطوير برمجية إكسبلويت لأي ثغرة يسهل إيجادها أو أي ثغرات أكثر تأثيرًا أكثر فعالية وكفاءة من حيث التكلفة.

[1] اعتبارًا من تاريخ توقف بياناتنا (1 آذار (مارس) 2016)

[2] حين يكتشف باحثان (أو أكثر) بشكل مستقل الثغرة نفسها، يُقال إن "اصطدامًا" قد حدث، ويُقال إن الثغرة قد "تداخلت". كما يشار في بعض الأحيان إلى معدل الاصطدام بمعدل التداخل.

الاستنتاج 1: قد يكون إعلان ثغرة حية (مجهولة للعامة) أو ميتة (معلومة للعامة) مضللًا أو مبسطًا للغاية

تكمن الممارسة الشائعة في تصنيف الثغرة ببساطة باعتبارها حية (مجهولة للعامة) أو ميتة (معلومة للعامة)؛ غير أن تحليلنا كشف عن وجود عدة فئات ثانوية لكل ثغرة، ما قد يجعل توصيف الثغرة باعتبارها حية أو ميتة مضللًا ومبسطًا للغاية.

قد تكون الثغرات الحية (المجهولة للعامة) هي تلك التي يتعقبها المدافعون بشكل ناشط – وتُسمى الثغرات "الحية" – أو تلك التي تبقى في منتج ما بشكل دائم إذ لم يعد البائع يحتفظ بالرمز أو يصدر تحديثات – وتُسمى الثغرات "الخالدة".

ويمكن الكشف عن الثغرات الميتة (المعلومة للعامة) بشكل علني من قبل الباحثين الذين اكتشفوا وجودها ("قضى عليها الباحثون") أو من قبل أي طرف آخر. في بعض الأحيان، يتمّ الكشف عن هذه الثغرات من خلال نشرة أمنية أو برنامج إصلاح أمني (قُضي عليها بواسطة "برنامج إصلاح أمني"). وفي أحيان أخرى، سيقوم المطورون أو الباحثون عن الثغرات بنشر خطأ أو ثغرة وجدوها على لائحة مرسلة عبر البريد الإلكتروني أو مدونة إلكترونية أو كتاب. قد يدرك الناشر أو لا يدرك أن الخطأ الذي تتم مناقشته هو في الواقع ثغرة أمنية، لذا ما من نشرة أمنية مرتبطة بالثغرة (قُضي عليها من خلال "مشاركتها علنًا").

لكن ثمة ثغرات أخرى هي شبه حية (مثل الزومبي)، لأنه يمكن استغلالها في نسخ قديمة لمنتج ما ولكن ليس في أحدث نسخة منه. وتتم إزالة ثغرات "عامل إعادة هيكلة الرمز" من خلال عمليات مراجعة الرمز، بدون اكتشافها أو الكشف عنها علنًا باعتبارها ثغرات أمنية.

ونظرًا إلى الطبيعة الديناميكية للثغرات، ما يمكن استغلاله اليوم لا يمكن استغلاله في اليوم التالي (والعكس صحيح).

وخلال التحقيق في حالة الثغرات، وجدنا أن أنظمة الثغرات الشائعة والانكشاف (CVE) لا توفّر دائمًا معلومات كاملة ودقيقة عن مدى خطورة الثغرات.

الاستنتاج 2: إن متوسط أمد الحياة المتوقع لبرمجيات إكسبلويت يبلغ 6.9 سنوات بعد اكتشافها الأولي، لكن حوالى 25% من برمجيات إكسبلويت لن تصمد لأكثر من عام ونصف في حين أن نسبة 25% أخرى ستصمد لأكثر من 9.5 سنوات

بعد الاكتشاف الأول من قبل باحث ثغرات، يبلغ متوسط أمد الحياة المتوقع لبرمجيات إكسبلويت 6.9 سنوات (تحديدًا 2521 يومًا)، ومن المرجّح أن تدوم أي برمجية إكسبلويت بين 5.39 و8.84 سنوات، علمًا بأن 25% فقط من الثغرات لا تصمد حتى لمدة 1.51 سنة، في حين تستمر 25% فقط أكثر من 9.5 سنوات. ويعني أمد الحياة الطويل نسبيًا عند 6.9 سنوات أن ثغرات يوم الصفر، لا سيما تلك التي تُعدّ برمجيات إكسبلويت لأجلها لاستعمالها من قبل الجيش أو الحكومة أو السوق، قديمة على الأرجح.

وفي حين أظهرت بياناتنا أن الحياة القصيرة تبلغ 1.5 سنة، قد تكون هذه الفترة طويلة بما يكفي بالنسبة إلى معظم باحثي الثغرات.

الاستنتاج 3: لم تشر خصائص ثغرة ما إلى حياة طويلة أو قصيرة، لكن قد تسعى تحليلات مستقبلية إلى مقارنة فيروس لينكس (Linux) مع أنواع أخرى من المنصات، ومراقبة أوجه الشبه بين رمز مصدر مفتوح وآخر مغلق وتجمعات مختلفة لنوع فئة برمجية إكسبلويت

بعد تقييم نوع الثغرة والمنصة المتضررة ونوع رمز المصدر ونوع فئة برمجية إكسبلويت، لم تظهر أي خاصية من الناحية الإحصائية باعتبارها "دليلًا" قد يشير إلى حياة قصيرة أو طويلة. وقد يُعزى ذلك سواء إلى نقص فعلي في الترابط أو نقص في القوة الإحصائية لرصد تلك الترابطات، نظرًا إلى العدد الصغير نسبيًا من الثغرات الميتة في مجموعة بياناتنا. وقد توفّر المزيد من البيانات نتائج أكثر أهمية من الناحية الإحصائية، بغض النظر عما إذا كان ذلك سيؤكد هذه النتيجة أو يجد أن خاصية محددة ومهمة غير واضحة.

وفي حين لم يكتسب أي أمر أهمية إحصائية، يوفر تحليلنا توجيهات بشأن أي فرضيات قد يكون من المفيد تجربتها في تحليلات مستقبلية، وبخاصة لمراقبة ديمومة ثغرات لينكس مقارنة بمنصات أخرى، ولتأكيد أوجه شبه ديمومة الثغرات الخاصة بنوع رمز المصدر المفتوح والمغلق، وللتحقيق في أهمية جمع برمجيات إكسبلويت من جانب العميل وبرمجيات إكسبلويت عن بعد معًا مقارنة بمجموعة برمجيات إكسبلويت محلية ومختلطة وغير ذلك.

وفي حال ظهور خصائص (يمكن تأكيدها أو دحضها بمزيد من البيانات)، يمكن إبلاغها إلى أولئك المنخرطين في البحث عن الثغرات أو عملية معالجة الثغرات أو الأمن عمومًا من أجل صقل ما يجب الاحتفاظ به وما يجب نشره بشكل علني.

الاستنتاج 4: اكتشفت أطراف أخرى نحو 5.7% من الثغرات لمجموعة محددة من ثغرات يوم الصفر بعد مرور سنة

يُعرف احتمال أن يكتشف طرفان مستقلان (أو أكثر) ثغرة ما بمعدل الاصطدام. (حين يكتشف باحثان أو أكثر بشكل مستقل الثغرة نفسها، يُقال إن "اصطدامًا" قد حدث، ويُقال إن الثغرة قد "تداخلت". كما يشار في بعض الأحيان إلى معدل الاصطدام بمعدل التداخل). وفي تحليلنا، تغيّرت معدلات الاصطدام بشكل كبير وفق الفاصل الزمني المستخدم (من 40% إلى أقل من 1%)، لذا يُعتبر توقيت "التخلص" من مجموعة ثغرات ميتة مهمًا. وقد وجدنا قيمة تداخل وسيطة تبلغ 5.76% (انحراف معياري بنسبة 6.79%) مقابل فاصل زمني لمدة 365 يومًا، وقيمة تداخل وسيطة تبلغ 0.87% (انحراف معياري بنسبة 5.3%) مقابل فاصل زمني لمدة 90 يومًا. وقد أسفر فاصل زمني لمدة 14 عامًا (أي جميع بياناتنا خلال فاصل زمني واحد) عن تداخل بنسبة 40%. وباستثناء الفاصل الزمني لمدة 14 عامًا، أظهرت بياناتنا معدل اصطدام منخفض نسبيًا. وقد يُعزى ذلك إلى أن الأطراف ضمن مجال الاستغلال الخاص يتطلعون إلى ثغرات مختلفة عن تلك التي يبحث عنها صائدو الثغرات لمشاركتها باعتبارها معرفة عامة، فضلًا عن استخدام تقنيات مختلفة لإيجاد الثغرات (على سبيل المثال غالبًا ما تكون الثغرات المكتشفة من خلال الاختبار العشوائي أو اختبار البرمجيات الآلي مختلفة عن تلك المكتشفة من خلال تحليل يدوي).

كما أظهرت البيانات أنه بين عاميْ 2002 و2016، حصل موت برمجيات إكسبلويت التي اندثرت (أي التي وجدتها أطراف مستقلة) بشكل سريع نسبيًا وغالبًا خلال السنة الأولى، رغم أن معدل الاكتشاف قد لا يكون متناسقًا كل عام (مثلًا اكتشفت الثغرات بعد 2008 بسرعة أكبر من تلك التي اكتُشفت قبل 2008).

وستُصقل نتائجنا بشكل إضافي بمعلومات أفضل حول كيفية حصول التقييم (أي الفاصل الزمني المستخدم من المنظمات والوكالات)، إضافةً إلى تحديد الثغرات التي تحتفظ بها جماعات خاصة أخرى.

الاستنتاج 5: فور اكتشاف ثغرة يمكن استغلالها، يكون الوقت الضروري لتطوير برمجية إكسبلويت فعالة بالكامل سريعًا نسبيًا، حيث يبلغ متوسطه 22 يومًا

تتفاوت مدة تطوير برمجيات إكسبلويت، لكنها تكون قصيرة نسبيًا عمومًا. وقد أظهرت بياناتنا أن 71% من برمجيات إكسبلويت تمّ تطويرها في غضون شهر واحد (31 يومًا أو أقل)، وحوالى الثلث (31.44 %) في غضون أسبوع أو أقل، في حين استغرق تطوير 10% منها فقط أكثر من 90 يومًا. واستغرقت أغلبية برمجيات إكسبلويت في مجموعة بياناتنا بين 6 و37 يومًا لتصبح فعالة بالكامل (بمتوسط 22 يومًا، بحدٍ أدنى يبلغ يومًا واحدًا وحدّ أقصى يبلغ 955 يومًا).

وقد تتوقّف تكلفة تطوير (وبالتالي قيمة أو سعر) برمجيات إكسبلويت على عدة عوامل تشمل: الوقت الضروري لاكتشاف ثغرة يوم صفر قابلة للنمو (وقت البحث)، والوقت اللازم لتطوير برمجية إكسبلويت للاستفادة من ثغرة يوم صفر (وقت تطوير برمجية إكسبلويت)، وتكلفة شراء جهاز أو رمز للمراجعة أو الاستحواذ عليه، والوقت الضروري لإعداد مختبر تجارب وتكلفة البنية التحتية أو الأدوات المناسبة والضرورية للاختبار والتحليل، والوقت اللازم لدمج برمجية إكسبلويت محددة في عمليات قائمة أخرى، ورواتب الباحثين المشاركين في عملية تطوير برمجيات إكسبلويت، ومزيج قاعدة الرموز (أي احتمال الاضطرار إلى مراجعة برمجية إكسبلويت وتحديثها لتناسب نسخًا جديدًا من الرمز من أجل الحفاظ على قدرة)، والعرض والطلب على برمجية إكسبلويت لمنصة أو قاعدة رموز معينة. وقد تتأتى القيمة الإضافية من فرادة ثغرة (مثلًا إن كانت الثغرة الوحيدة المكتشفة في منتج محدد) أو حاجة العميل وإطاره الزمني.

كان للثغرات المشتراة من أطراف ثالثة خارجية أمد حياة أقصر (متوسط 1.4 عام). وقد يشكل ذلك حجة لإيجاد ثغرات وتطوير برمجية إكسبلويت محليًا في حال الرغبة في تمديد الحياة.

لا يبدو أن وقت تطوير برمجية إكسبلويت يؤثّر على أمد حياة أو استمرارية هذه البرمجية.

وببساطة، يمكن لأي مهاجم خطير الحصول على ثغرة يوم صفر بسعر ميسور لأي هدف تقريبًا. غير أن تكاليف ملموسة أخرى (الاستحواذ على منتجات لإيجاد ثغرات فيها وإنشاء بنية تحتية للاختبار واستئجار مكان عمل، الخ) وتكاليف غير ملموسة (قيمة الطلب المرتفع ومنتج محدود العرض، الخ) قد تتسبب في ارتفاع الأسعار بشكل ملحوظ.

تداعيات الدفاع والهجوم

بما أن ثغرات يوم الصفر تتمتع بمتوسط أمد حياة متوقع يبلغ 6.9 سنوات، ويبدو أن التداخل بين ما تمّ الكشف عنه علنًا وما يتمّ اكتشافه سرًا صغيرًا نسبيًا، يكون للهجوم اليد العليا. كذلك،

وبما أنه لا يبدو أن أي من خصائص الثغرات تشير إلى حياة طويلة أو قصيرة، ومن شأن الإفراط في تبسيط الثغرات سواء باعتبارها حية (مجهولة للعامة) أو ميتة (معلومة للعامة) أن يقوّض مساعي الكشف عن الثغرة، يجب أن تركز الاستراتيجيات الأمنية على جميع أنواع الثغرات وليس على نوع واحد فقط.

فيحتاج المدافعون على الأرجح إلى خيارات أفضل لإيجاد ثغرات يوم الصفر ورصدها عندما يتمّ استغلال نظام ما أو حزمة برمجيات. كما أنه عوضًا عن التركيز فقط على اكتشاف ثغرات يوم الصفر، قد يتمكن المدافعون من ترجيح الكفة لصالحهم عبر البدء من فرضية المساومة والتحقيق في سبل تحسين تصميم هندسة النظام من أجل احتواء أثر المساومة واعتماد تقنيات مختلفة لتحديد الثغرات.

ويجدر بالمشاركين في التخطيط لعمليات هجومية باستخدام ثغرة يوم صفر محددة النظر في استخدامها في ظروف التخطيط القصيرة الأمد عمومًا. من ناحية أخرى، وبما أنه لا يبدو أن أي من خصائص الثغرات تشير إلى حياة طويلة أو قصيرة، قد يكون تخزين برمجيات إكسبلويت وتطويرها أكثر فعالية وكفاءة من حيث التكلفة لأي ثغرة يسهل إيجادها أو أي ثغرات أكثر تأثيرًا. ونظرًا إلى حياة الثغرات الطويلة ومعدل الاصطدام المنخفض، قد يكون وجود عدد قليل من الثغرات الاحتياطية كافيًا، وقد يتيح استخدام أي ثغرة يوم صفر لحزمة برمجيات محددة فترة زمنية أطول للتخطيط لعملية أو تنفيذها. فضلًا عن ذلك، وفي حالة محدودية استخدامها، قد تبقى الثغرات الخالدة أو التي خضعت لإعادة هيكلة رمزها قيّمة للعمليات، وذلك رهنًا بالنظام أو الهدف الذي تتواجد فيه: على الباحثين النظر في مراجعة الثغرات التي لم يعتبروها سابقًا قابلة للاستغلال بشكل منتظم.

وببساطة، يمكن لأي مهاجم خطير الحصول على ثغرة يوم صفر بسعر ميسور لأي هدف تقريبًا. ولا تتأتى غالبية تكلفة برمجيات إكسبلويت لاستغلال ثغرات يوم الصفر من العمالة بل من القيمة المتجذرة فيها وغياب العرض. ويمكن أن تتسبب تكاليف ملموسة أخرى (الاستحواذ على منتجات لإيجاد ثغرات فيها وإنشاء بنية تحتية للاختبار والاحتفاظ ببرمجية إكسبلويت وتحويلها لتعمل على عدة نسخ واستئجار مكان عمل، الخ) وتكاليف غير ملموسة (قيمة الطلب المرتفع ومنتج محدود العرض، الخ) في ارتفاع الأسعار بشكل ملحوظ. ويجدر بالمدافعين التنبّه إلى التوافر والاقتصاديات ليتمكنوا من الدفاع بفعالية أكبر.

هل يجب تخزين الثغرات أم لا؟

قد تقرر الحكومات إبقاء ثغرات يوم الصفر سرية، سواء لأغراض دفاعية (مثلًا اختبار الانتشار) أو للعمليات الهجومية. ويتطلب قرار التخزين أخذ عدة عوامل في الحسبان بدقة، بما في ذلك الثغرة بحدّ ذاتها واستخدامها وظروف استخدامها وخيارات أخرى قد تكون متاحة لتحقيق النتيجة المرجوة.

وتُظهر تحليلاتنا أن ثغرات يوم الصفر قد تتمتع بمتوسط حياة طويل ومعدلات اصطدام منخفضة. وقد يشير التداخل المحدود إلى كثافة الثغرات (أي وجود ثغرة مختلفة أخرى بشكل معتاد) أو صعوبة اكتشافها (حيث إن هاتين الخاصيتين ليستا حصريتين بالضرورة). في حالٍ

وجود ثغرة أخرى بشكلٍ معتاد، قد يُعتبر مستوى الحماية الذي يحصل عليه المستخدمون من الباحثين الذين كشفوا ثغرة ما متواضعًا، وقد يستنتج البعض أن تخزين ثغرات يوم الصفر قد يكون خيارًا منطقيًا. في حال كان يصعب اكتشاف ثغرات يوم الصفر، حينها قد يدعم الاحتمال الضعيف بأن يجد آخرون الثغرة نفسها حجة الاحتفاظ بمخزون.

من ناحية أخرى، يُظهر تحليلنا أن معدلات الاصطدام لثغرات يوم الصفر ليست معدومة. وقد يشير البعض إلى أنه في حال بروز أي احتمال أن يكتشف شخص آخر (بخاصة منافس) ثغرة يوم الصفر نفسها، تبرر عندها التداعيات الخطيرة المحتملة لإبقاء ثغرة يوم صفر سرية وترك العامة معرضين للخطر الكشف الفوري عن الثغرة وإصلاحها. ومن هذا المنطلق، قد يتمثّل القرار الأفضل بالتخزين فقط في حال كان المرء واثقًا من أن أحدًا غيره لن يكتشف ثغرة يوم الصفر، وإلا من الأفضل الكشف عنه.

شكر وعرفان

يعتمد إعداد دراسات RAND عادةً على مجموعة كبيرة من الداعمين والمساهمين والمساعدين. نوّد في بادئ الأمر أن نشكر المساهمات الخيرية من داعمي RAND والدخل المولّد من العمليات ومعهد RAND للعدالة المدنية على الدعم السخي الذي جعل إعداد هذا البحث ممكنًا. لقد قدّم المراجعان لدينا، ألن فريدمان (Allan Friedman)، مدير مبادرات الأمن الإلكتروني لدى إدارة الاتصالات والمعلومات الوطنية التابعة لوزارة التجارة الأمريكية، وأخيل شاه (Akhil Shah)، أحد كبار المهندسين لدى مؤسسة RAND، ملاحظات واقتراحات حكيمة ساهمت في تحسين هذا التقرير بشكل كبير. كما قدّم الزملاء في RAND مارتين ليبيكي (Martin Libicki) وسينتيا ديون-شوارتز (Cynthia Dion-Schwarz) وكاوليون أوكونيل (Caolionn O'Connell) وإيغور ميكوليك-توريرا (Igor Mikolic-Torreira) وساشا رومانوسكي (Sasha Romanosky) رؤى وملاحظات قيّمة. كما نتقدّم بالشكر إلى إرنستو أمارال (Ernesto Amaral) على خبرته ومساهمته بشأن منهجيات تقييم البيانات الديمغرافية، وأندريا غولاي (Andrea Golay) على أفكارها بشأن التصميم الأولي، وديفيد كرافيتز (David Kravitz) على تحليله الإحصائي. فضلًا عن ذلك، شكّلت كريستين ليو (Christine Liu) مصدرًا رائعًا لمنهجيات أمد الحياة. وخصصت ميليسا بومان (Melissa Bauman) الكثير من وقتها للمساعدة في الدفق والنماذج وكذلك قدّمت إريكا روبلز (Erica Robles) مساعدة كبيرة في إعداد المستند.

نقدّر حقًا هؤلاء الخبراء الذين خصصوا الوقت لمشاركتنا أفكارهم وتجاربهم، ولتقديم مساهماتهم والانخراط في مناقشات مشوّقة ومحفزة حول ثغرات يوم الصفر. ومن بين هؤلاء الخبراء نذكر دان غير (Dan Geer) وكايتي موسوريس (Katie Moussouris) وتوماس دوليان (Thomas Dullien) ومارا تام (Mara Tam) ودان غيدو (Dan Guido) ووندي ناثر (Wendy Nather) وستيف كريستي كولي (Steve Cristey Coley) وتراي هير (Trey Herr) وأمالي كورنينغ ويدج (Amalie Korning Wedege) وبرايسون بورت (Bryson Bort) إضافةً إلى ثمانية أفراد من مطوري برمجيات إكسبلويت والباحثين عن الثغرات والوسطاء طلبوا عدم نسب أي فضل إليهم.

لا تعني المساعدة والرؤى التي حصلنا عليها من الأشخاص الذين شكرناهم أعلاه أنهم يوافقون على الآراء المعبّر عنها في هذا التقرير. نحن وحدنا مسؤولون عن محتوى التقرير، بما في ذلك أي أخطاء أو هفوات.

أخيرًا، نودّ أن نعبّر عن امتناننا الكبير لـBUSBY على المعلومات.

الفصل الأول

مقدمة

يستمر النقاش السياسي حول ما إذا كان يتعين على الحكومة الأمريكية، أو أي حكومة أخرى، الاحتفاظ بما يسمى ثغرات يوم الصفر أو الكشف عنها لكي يصبح بالإمكان إصلاحها. فالذين لديهم علم بثغرات يوم الصفر قد يعدّون "برمجيات إكسبلويت"، أي مجموعة رموز تستغل الثغرة، من أجل ولوج أجزاء أخرى من النظام أو تطوير رمز خاص بهم أو التصرف باعتبارهم مديرين أو إتمام بعض الخطوات الأخرى. لكن الكثيرين يخشون من أن إبقاء هذه الثغرات سرية قد يعرّض الأشخاص الذين يستخدمون هذه البرمجية الهشة لهجمات خبيثة ومحاولات أخرى لجمع معلوماتهم الشخصية.

وغالبًا ما يؤجج النقاش حول ما إذا كان يجب الاحتفاظ بهذه الثغرات أو الكشف عنها مدى التداخل المحتمل بين ثغرات يوم الصفر أو برمجيات إكسبلويت التي تحتفظ بها الحكومة الأمريكية وتلك التي يخزنها خصومها. إذا كان للطرفين المخزونات نفسها، يرى البعض أنه لا جدوى من إبقائها خاصة، حيث أن تداخلًا أصغر قد يبرّر الاحتفاظ بها. لكن في ظل غياب المعلومات حول التداخل أو أي مؤشرات ملموسة تستند إلى بيانات فعلية، يصعب اتخاذ قرار مستنير بشأن التخزين.

ومن أجل معالجة هذه المسألة، كان لمؤسسة RAND نفاذ نادر إلى مجموعة من المعلومات الغنية حول ثغرات يوم الصفر وبرمجيات إكسبلويت. نستعرض في هذه الدراسة مجموعة البيانات ونكشف عن معلومات معمّقة حول القطاع ونعد بعض المؤشرات الأولية المرتبطة بحالة ثغرات يوم الصفر وبرمجيات إكسبلويت وديمومتها واحتمال أن يكتشفها الآخرون (ما يعرف بمعدل الاصطدام"). كما نتطرق إلى وقت العمل الضروري لإنشاء برنامج إكسبلويت لاستغلال ثغرة يوم صفر.

معلومات قليلة متوافرة حول مدى انتشار ثغرات يوم الصفر أو استعمالها أو فوائدها أو أضرارها

يهدف بائعو البرمجيات إلى تصميم برمجيات خالية من العيوب، لكنه هدف طموح. فالأجهزة والشبكات والأنظمة والعمليات بحدّ ذاتها تحتوي على عدد من الأخطاء. وتتراوح التقديرات بين 3 و20 خطأ لكل 1000 خط من الرموز، وأقل بنسبة واحدة أو اثنتين عقب مراجعة دقيقة (ماكونل McConnell، 2004). ويختلف عدد الأخطاء بحسب التطبيق أو الجهاز أو الأسلوب (ماكونل، 2004).

والثغرة هي نوع من الأخطاء التي تؤدي إلى ضعف أمني في تصميم أو تطبيق أو تشغيل نظام أو تطبيق (المجلس الوطني للأبحاث NRC، 1999).[1] يمكن إدخال الثغرات ونقاط الضعف عمدًا، أي من خلال سلوك مقصود، أو عن غير قصد، من خلال خطأ عرضي في التصميم أو التطبيق (المجلس الوطني للأبحاث، 2009). أما برمجية إكسبلويت، فهي "رمز خبيث يستغل ثغرات البرمجية من أجل إفساد الحاسوب أو تعطيله أو السيطرة عليه بدون موافقة المستخدم، وعادةً بدون معرفته" (مايكروسوفت Microsoft، 2013). ولكن لا تعتبر جميع الأخطاء بمثابة ثغرات، ولا يمكن استغلال جميع الثغرات بشكل مفيد.[2] وقد تمكّن بعض الثغرات مهاجمًا ما من تجاوز صلاحيات تنفيذ هجوم قطع الخدمة، في حين تسمح أخرى فعليًا لمهاجم ما بالحصول على ميزة تنفيذ رمز عن بعد، ما يُعتبر غالبًا الهدف الأسمى، حيث يشغل النظام المعرّض للخطر رمز المهاجم بدون معرفة المستخدم.

وثغرات يوم الصفر (أو ثغرة يوم الصفر) هي ثغرات لم يتم نشر أي طريقة لإصلاحها أو تعديلها بشكل علني. في بعض الحالات، قد لا يكون بائع البرمجيات على دراية بوجود الثغرة (يشير مصطلح ثغرة يوم الصفر إلى عدد الأيام التي علم خلالها بائع البرمجية بوجود ثغرة (ليبيكي Libicki، أبلون Ablon، وويب Webb، 2015).[3] وتمثّل برمجية إكسبلويت لاستغلال ثغرات يوم الصفر رمزًا يستغل ثغرة يوم الصفر ويسمح لمستحدث برمجية إكسبلويت بولوج أجزاء أخرى من النظام أو إعداد رمزه/ها الخاص أو التصرّف بصفته مديرًا أو القيام بأفعال أخرى قد تلحق الضرر.[4] يمكن أن تكون برمجية إكسبلويت لاستغلال ثغرات يوم الصفر مفيدة للغاية لمن يصبح

[1] يمكن أن تشمل الثغرات أيضًا تهيئة خاطئة ونقصًا في الضوابط المناسبة.

[2] على سبيل المثال، قد لا يتوافر دومًا مسارًا للرمز من الثغرة إلى جزء مفيد من الرمز. وقد يكتشف المرء مثلًا تجاوزًا لسعة المخزن المؤقت، لكن ما من مسار يسمح بممارسة هذا التجاوز. أو قد يكتشف ثغرة تلاعب بمسار، حيث بإمكان مهاجم الاستحواذ على ملف من موقع ليس من المفترض به ولوجه (ويتمثل مثل شائع في إعطاء أمر للوصول إلى دليل لا يمكن للمهاجم ولوجه). لكن لا يعني وجود ثغرة (أي حيث يمكن للمهاجم التنقل في نظام الملف) أن التطبيق متاح (ربما ثمة ملف ثابت/جامد لا يشكّل جزءًا من الدليل، وبالتالي، لن يتمكن المهاجم من دخول موقع الملف المرجو بأي شكل من الأشكال) (ليبيكي، أبلون وويب، 2015).

[3] غالبًا ما يشير مصطلح ثغرة يوم الصفر إلى ثغرات يوم الصفر وبرمجيات إكسبلويت لاستغلال ثغرة يوم الصفر – برمجيات إكسبلويت المستحدثة للاستفادة من ثغرات يوم الصفر. وفي هذا التقرير، سعينا إلى التمييز بين الحديث عن ثغرة والحديث عن برمجية إكسبلويت.

[4] في بعض الأحيان، يتمّ الخلط بين برمجيات إكسبلويت والمغروسات (implants). في هذا التقرير، نركّز على برمجيات إكسبلويت وتطويرها. ثمة بعض جوانب الاختلاف الملحوظة بين برمجية إكسبلويت ومغروس: فبرمجية إكسبلويت توفّر ولوجًا أوليًا، وغالبًا أيضًا القدرة على تنفيذ رمز من خلال الاستفادة من بعض الثغرات في عملية نظام ما، ومن ثم تسهيل المغروس أو تحميله، ما يقوي هذا الولوج ويحافظ عليه (أي يحقق الاستمرارية)، ويكون له بعض التأثير على النظام (ليس مجرد عملية في النظام). على سبيل المثال، نظرًا إلى تمكّن جهة فاعلة من تشغيل نظام باعتبارها مستخدمًا، قد تعدّل برمجية إكسبلويت لتجاوز الصلاحيات عملية ما للحصول على صلاحيات أعلى، ما يسمح بالتالي لمغروس باستخدام حمولة للقيام بشيء ما بهذه الصلاحية الإضافية. بعبارة أخرى، توفر برمجيات إكسبلويت ولوجًا وتخوّل جهة فاعلة من القيام بـ"شيء". أما المغروسات فهي المسؤولة عن القيام بهذا "الشيء". وقد تكون هذه الخطوط غامضة أحيانًا، لا سيما في حالة الاستغلال الآلي.

على دراية بها، نظرًا إلى أن كل نظام يشغّل البرمجية يحمل الثغرة وإلى أنه يصعب رصدها ووقفها.[5]

يستخدم المهاجمون ثغرة يوم الصفر من أجل السعي وراء منظمات وأهداف تواظب على مواكبة برامج الإصلاح. أما الذين يهملون الأمر فيمكن أن يتعرضوا لهجوم عبر الثغرات التي لديها برامج إصلاح لكن لم يتمّ تطبيقها. وعليه، تكون ثغرات يوم الصفر وبرمجيات إكسبلويت مفيدة في العمليات الإلكترونية، سواء تلك التي يشنها المجرمون أو الجيوش أو الحكومات، وكذلك في المجالات الدفاعية (مثلًا اختبار الانتشار) والأكاديمية. وحتمًا، بدأت نماذج الأعمال والأسواق بالاستعداد للمداولة أو بيع ثغرات يوم الصفر القابلة للاستغلال.

وفي حين تنطوي العديد، وربما معظم، الاختبارات الدفاعية أو العمليات الهجومية على ثغرات معروفة (أي ليس ثغرة يوم صفر)، استُخدمت برمجيات إكسبلويت لاستغلال ثغرات يوم الصفر في بعض الحالات المهمة، وأشهرها دودة ستوكسنت (Stuxnet)، التي اعتمدت على أربع ثغرات يوم صفر من مايكروسوفت لإلحاق الضرر ببرنامج إيران النووي (نارين Naraine، 2010). وفي حالة أخرى، شكّلت هارتبليد (Heartbleed) ثغرة خطيرة في أوبن إس إس إل (OpenSSL) (وهي مكتبة تشفير تستخدمها ملايين المواقع) قد تسمح بتسريب مفاتيح خاصة ومشاركتها مع المهاجم. وبالرغم من أنه ما من حالة معروفة عن مهاجمين يستغلون ثغرة هارتبليد في العمليات، يشكّك الكثيرون في أنه تم استخدامها في بعض العمليات بينما كانت لا تزال ثغرة يوم صفر.

وناهيك عن هذه الأمثلة المهمة، ما من معلومات كثيرة متوافرة حول مدى انتشار ثغرات يوم الصفر أو استعمالها أو فوائدها أو أضرارها بشكل فعلي. وغالبًا ما تكون المحادثات تخمينية أو مبنية على ما تم اكتشافه بعد استغلال الثغرة ورصدها في هجوم ما.[6] وفي حين تم إعداد بعض النماذج الاستدلالية للتحقق من معدل استنزاف ثغرات البرمجيات (ليبيكي، أبلون وويب، 2015) ونتائج نظرية الألعاب الخاصة بالتخزين (مور Moore وفريدمان Friedman وبروكاتشيا Procaccia، 2010)، ومعدل الاصطدام في مختلف قواعد الرموز (موسوريس Moussouris وسيغيل Siegel، 2015)، والاقتصاديات العامة لثغرات البرمجية (كوين Kuehn ومولر Mueller، 2014)، ما من بحث متاح علنًا بشأن هذا الموضوع قد بُني على بيانات فعلية حول ثغرات يوم الصفر الحالية. نقدّم في هذا التقرير رؤى قائمة على البيانات بشأن ثغرات يوم الصفر يمكنها دعم الأمثلة البديلة التقليدية وآراء الخبراء، واستكمال الجهود الحالية لوضع إطار عمل لاتخاذ قرار بشأن الكشف عن ذاكرة ثغرات يوم الصفر وبرمجيات إكسبلويت أو الاحتفاظ بها، وتوفير المعلومات اللازمة لمناقشات السياسة الجارية المحيطة بثغرات يوم الصفر في ما يخص تخزين الثغرة والكشف عنها.

[5] مع ذلك، يمكن لنظام يعاني ثغرة أن يتجنّب استغلاله في حال وجود برمجية أو معدات حاسوبية في مكان آخر من النظام تحول دون تداعيات برمجية إكسبلويت أو تحدّ منها.

[6] إنها على الأرجح مجموعة فرعية من جميع برمجيات إكسبلويت المستخدمة، على افتراض أنه لا يمكن رصد جميع برمجيات إكسبلويت المستخدمة.

هل يجدر بالحكومة الأمريكية الكشف عن ثغرات يوم الصفر؟

يستمر النقاش السياسي حول ما إذا كان يتعين على الحكومة الأمريكية، أو أي حكومة أخرى، الاحتفاظ (أو تخزين) بثغرات يوم الصفر لاستخدامها الخاص أو الكشف عنها للبائعين المتضررين من أجل إصلاحها. فعندما تقوم جهة بتخزين الثغرات وبرمجيات إكسبلويت لاستخدامها الخاص، تكون أنظمة أخرى تستخدم البرمجية نفسها عرضة للتأثر (بما فيه أجهزة كمبيوتر وأنظمة المواطنين والموظفين الحكوميين والبنية التحتية الرئيسية). ويخشى البعض أنه من خلال تخزين الثغرات وعدم تحذير البائعين المتضررين، سيجد الخصوم أو المجرمون الثغرات نفسها ويستخدموها لمآرب شخصية.

ويتمثل العاملان المهمان لهذا النقاش في:

1. **ديمومة الثغرة:** إلى متى يبقى البائع أو العامة غافلين عن وجود الثغرة.[7]
2. **معدل الاصطدام:** احتمال أن تجد جهة أخرى بشكل منفصل ثغرة يوم صفر سبق لجهة أخرى أن اكتشفتها.[8]

لا يجوز اعتبار التخزين مسألة واضحة ("احتفظ بها جميعها" أو "اكشفها جميعها") إذ تدخل العديد من الاعتبارات والفروق في الحسبان. في حال وجد خصم ثغرة يوم صفر، قد يكون تخزين هذه الثغرة من قبل حكومته مضرًا: فليس باستطاعة الخصم التحوط ضد استخدامها من قبل أي شخص فحسب، بل قد يعرّض شريحة كبيرة من السكان إلى الخطر. فضلًا عن ذلك، تحدث هوية مكتشف ثغرة يوم الصفر فرقًا، على سبيل المثال، بائع متضرر أو صائد أخطاء يعمل بالنيابة عن البائع، أو مجرمو الإنترنت أو غيرهم من الجهات الفاعلة العدوانية الذين يخفون المعلومات عن البائع والعامة.

من ناحية أخرى، قد يحدّ إفصاح الحكومة عن جميع الثغرات بشكل كبير أي ميزة تتمتع بها في استخدام ثغرات يوم الصفر لاختبار دفاعي أو عمليات هجومية. قد توجد فئات من الثغرات أو بعض المنتجات التي يكون من المنطقي الكشف عن أي ثغرات مكتشفة فيها، وأخرى لا نفع من كشف ثغراتها.[9]

---

[7] هنا ننظر ببساطة إلى الإطار الزمني بين جهة تكتشف ثغرة يوم صفر بشكل سري وأخرى تمّ الإفصاح عنها للعامة. لا نأخذ في الحسبان الوقت الذي قد يستغرقه البائع لإصلاح الثغرة فور الكشف عنها.

[8] حين يكتشف باحثان (أو أكثر) بشكل مستقل الثغرة نفسها، يُقال إن "اصطدامًا" قد حدث، ويُقال إن الثغرة قد "تداخلت". كما يشار في بعض الأحيان إلى معدل الاصطدام بمعدل التداخل. ويحدث اصطدام قريب عندما يجد الباحثون ثغرة مختلفة في الموقع نفسه من الرمز. انظر الملحق D، اصطدامات قريبة" للمزيد من المعلومات حول الاصطدامات القريبة.

[9] على سبيل المثال، قد يجد الباحثون عن الثغرات بسهولة تجاوزات سعة المخزن المؤقت نفسها، لكن يميل كل واحد منهم إلى إيجاد ثغرات مختلفة للتلاعب بالمسار، ما يبرر الإفصاح عن واحدة دون الأخرى. على نحو مماثل، ربما يستخدم المواطنون أو أصحاب الشأن في دولة ما عمومًا منتجات من مصنّع أمن شبكات واحد، وقد يرغبون في الإفصاح عن الثغرات الموجودة في منتجات ذلك المصنّع، إنما ليس تلك المكتشفة في منتجات منافس.

ثمة العديد من الاعتبارات التي يرغب أصحاب الشأن بمعالجتها

كان كل من الوكالات الحكومية وبائعي برامج الحماية الأمنية والباحثين المستقلين يحاولون تحديد ثغرات يوم الصفر التي يجب التمسك بها وإلى أي مدى. ويشمل ذلك عمومًا فهم (1) أرجحية الصمود وأمد الحياة المتوقع لثغرات يوم الصفر وبرمجيات إكسبلويت (الديمومة) و(2) احتمال أن تجد جهة أخرى بشكل منفصل ثغرة يوم صفر سبق لجهة أن اكتشفتها (معدل الاصطدام). وفي حين قد تكون ديمومة ثغرة خيارًا واضحًا على المقياس المرغوب فيه، يكتسب معدل الاصطدام أهمية بدوره، إذ قد يشير التداخل إلى نسبة المخزون التي وجدها شخص آخر، وربما أنواع الثغرات التي قد يكون مرغوبًا في تخزينها.[10]

وإلى حدّ ما، يهتم أصحاب الشأن أيضًا بمعرفة تكلفة إيجاد ثغرات وتطوير برمجيات إكسبلويت موثوقة، وكذلك أسعار مشترياتهم. وتشمل اعتبارات أخرى مدة احتفاظ نظام مستهدف بتشكيله الحالي ونسخة رمزه وتواتر إصلاحه ومدة الاحتفاظ بثغرة ما قبل الكشف عنها للعامة ومن هو الهدف.

لقد دارت عدة نقاشات حول هذا الموضوع: عقب اكتشاف ثغرة هارتبليد والكشف عنها في 2014، حدد مايكل دانيال (Michael Daniel)، منسق الأمن الإلكتروني في البيت الأبيض بعض الاعتبارات.[11] وناقش آخرون فوائد الإفصاح وتحدياته مقابل الاحتفاظ بالثغرة (شناير Schneier، 2014)، واكتشفوا أسواقًا مختلفة (فيدلر Fidler وغرانيك Granick وكرينشو Crenshaw 2014؛ كوين ومولر، 2014؛ ليبيكي وأبلون وويب، 2015)، وتحققوا من عملية معالجة الثغرات (شوارتز Schwartz ونييك Knake، 2016)، ودققوا في دور الإفصاح في تحسين الأمن أو تقويضه (رانسبوثام Ransbotham ومايتر Mitre، 2011)، وراجعوا التوصيات المحددة المتعلقة بسبل استغلال البرمجيات من مجموعة المراجعة التابعة للرئيس الأمريكي (كلارك Clark وآخرون، 2013)، وعقدوا حلقات نقاش خاصة بهذا الموضوع (زيتر Zetter، 2015)، وأبدوا آراءهم بشأن ما إذا كان احتفاظ الحكومة بثغرات يوم الصفر يضعف الأمن الرقمي (كروكر Crocker، 2016).

[10] تجدون نقاشًا أكثر تعمقًا حول التداخل في مرحلة لاحقة من هذا الفصل.

[11] حدد مايكل دانيال، منسق الأمن الإلكتروني السابق في البيت الأبيض، بعض الاعتبارات (دانيال Daniel، 2014):
- ما مدى استخدام النظام الهش في بنية الإنترنت التحتية الأساسية و/أو في أنظمة البنية التحتية المهمة الأخرى و/أو في الاقتصاد الأمريكي و/أو في أنظمة الأمن القومي؟
- هل تطرح الثغرة، في حال عدم إصلاحها، مخاطر كبيرة؟
- ما مدى الضرر الذي قد تلحقه دولة منافسة أو مجموعة إجرامية على معرفة بهذه الثغرة؟
- ما هو احتمال أن نعرف إن كان شخص آخر يستغل هذه الثغرة؟
- إلى أي مدى نحتاج إلى المعلومات التي نعتقد أنه يمكننا الحصول عليها عبر استغلال هذه الثغرة؟
- هل من سبل أخرى للحصول عليها؟
- هل يمكننا استخدام الثغرة لفترة وجيزة قبل الإفصاح عنها؟
- ما هو احتمال أن يكتشف شخص آخر الثغرة؟
- هل يمكن إصلاح الثغرة أو الحد من خطورتها؟

المسائل البحثية والغرض من هذا البحث

يعتمد قرار تخزين الثغرات على الأرجح على عدة عوامل مستقلة عن الثغرات بحدّ ذاتها (مثلًا ما نوع منظمتنا؟ كيف يمكن للاحتفاظ بالثغرات أن يُلحق الضرر بنا أو مساعدتنا؟). عندما ينتقل النقاش إلى البحث في ثغرات محددة، قد تُطرح أسئلة محددة، منها:

1. **الحالة:** هل الثغرة هي ثغرة يوم صفر حقًا؟ هل هي "حية" (مجهولة للعامة) أم "ميتة" (معلومة للآخرين)؟
2. **الديمومة:** إلى متى ستبقى الثغرة غير مكتشفة وغير معلنة؟
3. **معدل الاصطدام:** ما هو احتمال أن يكتشف آخرون الثغرة ويفصحون عنها (بما يشمل الباحثين الخاصين الآخرين والبائعين المتضررين)؟[12]

قد يكون من المفيد أيضًا معرفة:

4. **التكلفة:** ما هي تكلفة تطوير برمجية لاستغلال الثغرة (مثلًا بغية المساعدة على تحديد سعر الشراء)؟

قد تكون الإجابة عن هذه الأسئلة مفيدة لتحديد الثغرات التي يجب تخزينها وتلك التي يجب إصلاحها والكشف عنها للبائع.[13] وفي حين تبرز مؤشرات على وجود معايير عالية المستوى لاتخاذ قرار التخزين أو الاحتفاظ بالثغرات (مثلًا دانييلز Daniels، 2014)، ما من مقاييس متوافرة علنًا أو بيانات صلبة لتوفير المعلومات الضرورية للنقاش، ما يصعّب عملية اتخاذ قرار مستنير بشأن التخزين.

وتهدف هذه الدراسة إلى وضع بعض المقاييس الأولية في ما يخص ثغرات يوم الصفر من أجل المساعدة على توفير المعلومات الضرورية للنقاش. وبشكل خاص، نقدّم في هذا التقرير بعض المقاييس الأولية على حالة ثغرات يوم الصفر الحقيقية وبرمجيات إكسبلويت وديمومتها ومعدل اصطدامها عبر استخدام تطبيقات حديثة لأساليب إحصائية تقليدية، وهي منهجية يمكن تكرارها في مجموعات أخرى من بيانات الثغرات وبرمجيات إكسبلويت. كما نتطرق إلى بعض التكاليف والجهود الضرورية لاستحداث برمجية إكسبلويت من حيث وقت العمل الضروري.

[12] نستخدم في هذا التقرير مصطلح اكتشاف. غير أن البعض قد يرون أن استخدام مصطلح تحديد أو تعقّب هو أكثر ملاءمةً (مثلًا داي زوفي Dai Zovi، 2016). نقرّ بالفوارق لكننا نصرّ على مصطلح اكتشاف.

[13] يستند قرار الإفصاح عنها للعامة ولكن ليس للبائع (ما يسمى "الإفصاح الكامل" عوضًا عن "الإفصاح المنسق") إلى مجموعة مختلفة من الاعتبارات على غرار مدى تعنّت البائع أو مدى الانتشار العلني الذي يرغب فيه المكتشف.

الجمهور المستهدف من هذا البحث

يتألف الجمهور الرئيسي المستهدف من هذا البحث من صناع السياسات الذين يسعون إلى اتخاذ أفضل القرارات الممكنة بشأن كيفية الحدّ من الثغرات في البلاد مع الحفاظ في الوقت نفسه على خيارات متينة للعمليات الإلكترونية.

قد يزود هذا البحث بائعي البرمجيات/الباحثين عن الثغرات وصناع السياسات بالمعلومات الضرورية من خلال تسليط الضوء على التداخل بين الثغرات المكتشفة بشكل خاص وعلني، مع إبراز خصائص هذه الثغرات وتوفير نظرة معمّقة على تطوير برمجيات إكسبلويت لاستغلال ثغرات يوم الصفر.

لا بدّ من أن يكون هذا البحث مهمًا للمنخرطين في عملية معالجة الثغرات، وموظفي وزارة الدفاع الأمريكية المسؤولين عن الاستحواذ على ثغرات يوم الصفر وتطوير سياسات تتعلق بثغرات البرمجيات، ووزارة الأمن الداخلي الأمريكية (نظرًا إلى عملها على أنظمة كشف التسلل (IDS) مثل نظام أينشتاين Einstein III 3)، ووزارة الخارجية الأمريكية (نظرًا إلى دورها في مسائل ضبط الصادرات)، ومكونات الخدمة العسكرية المنخرطة في المفاعيل الإلكترونية، ومكتب التحقيقات الفيدرالي (الذي يكتشف ويلاحق النشاط الإجرامي المعتمد على ثغرات يوم الصفر والذي يستخدم هذه الثغرات لإجراء تحقيقات جنائية)، ومختلف المكاتب الحكومية المشاركة في إيجاد سياسات ذات صلة ببرمجيات إكسبلويت لاستغلال ثغرات يوم الصفر أو تطويرها أو استخدامها.

ومن بين المجموعات الأخرى التي قد تهمها نتائج البحث:

- المجموعات التي تبحث بنشاط عن الثغرات وتقوم بتطوير برمجيات إكسبلويت لأغراض هجومية. قد تساعدها النتائج على تعزيز كفاءتها في تطوير برمجيات إكسبلويت لاستغلال الثغرات التي تتمتع بأمد حياة متوقع طويل، وأن تكون أكثر اقتصادًا في تطوير برمجيات إكسبلويت لا تتطلب الكثير من الوقت (أي زيادة معدل الفائدة/التكلفة) واستحداث نماذج تسعير معيارية.
- المجموعات التي تطور برامج مكافئات أمنية (bug-bounty) أو تشتري ثغرات أو برمجيات إكسبلويت. قد تساعدها النتائج على تعديل نماذج أعمالها لتتواءم مع الجهود المبذولة لاكتشاف مختلف أنواع الثغرات واستغلالها وانتقاء الثغرات التي قد تدوم لأطول فترة قبل أن يكتشفها الآخرون.
- المجموعات التي تطوّر منتجات أمن البرمجيات أو التي تبحث بنشاط عن الثغرات لأهداف دفاعية. قد توفر لها النتائج رؤى حول أنواع الثغرات أو فئات برمجيات إكسبلويت التي قد تبقى غير مكتشفة لأطول وقت، ما يمكن استخدامها لتركيز جهود التطوير الآمن أو مؤشرات الثغرات التي تختبر تدابير دفاعية جديدة.

تجزئة مجال ثغرة يوم الصفر

تعتبر ثغرات يوم الصفر وبرمجيات إكسبلويت قيّمة للعديد من المجتمعات المختلفة (الدوائر العسكرية والدفاعية، وبائعو البرمجيات، والباحثون عن الثغرات) وهي مفيدة لإجراء عمليات هجومية وتعزيز التدابير الدفاعية على السواء.

تبحث مجموعات مختلفة عن ثغرات يوم الصفر، منها: الوكالات الحكومية ومقاولوها في مجال الدفاع، وبائعو البرمجيات والباحثون في مجال الأمن، سواء باعتبارهم مقاولين مستقلين أو موظفين في شركة أبحاث تُعنى بالثغرات.

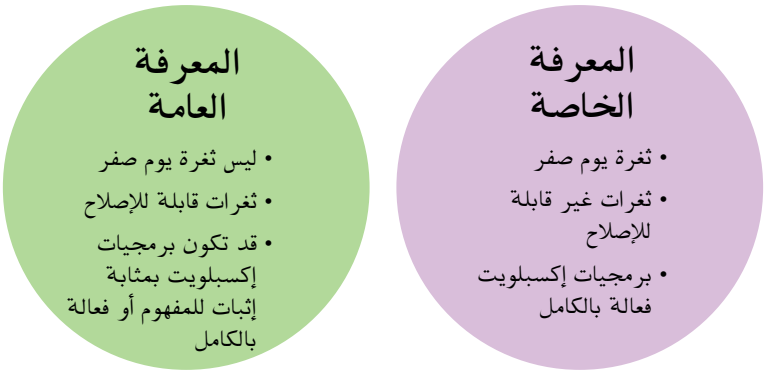
ويمكن للمرء أن يميّز بين الذين يبحثون عن ثغرات يوم الصفر لاستغلالها وإبقائها خاصة والذين يستهدفون إصلاح ثغرات يوم الصفر والإفصاح عنها (علنًا). ويختلف تكوين الأشخاص وعددهم في كل مجموعة تمامًا كمخزون ثغرات يوم الصفر ومصادر التمويل.

مخزون ثغرات (يوم الصفر)

يتمثل الفهم المشترك في وجود نوعين من الثغرات: (1) التي يحتفظ بها للاستعمال الخاص و(2) التي تشكّل جزءًا من المعرفة العامة (راجع الشكل 1.1). وغالبًا ما يُعتقد أن الثغرات في المجال الخاص غير قابلة للإصلاح وتُستخدم حصرًا لأغراض هجومية. أما الثغرات في المجال العام، فغالبًا ما يُفترض أن البائع يعلم بوجودها حيث أنه يمكن إصلاحها. وما إن يتمّ الكشف عن ثغرة إلى العلن، لا يمكن اعتبارها تمثل ضررًا محتملًا للعامة.

لكن هذه النظرة مبسّطة بشكل مفرط عمومًا. ويمكن أن تبقى الثغرات خاصة بينما يجري إصلاحها بسرية (فلا تعد تعتبر ثغرة يوم صفر). في المقابل، قد تكون الثغرات جزءًا من المعرفة العامة لكنها تبقى بدون إصلاح. وضمن مجال المعرفة الخاصة، قد تبرز ثغرات يعرفها

الشكل 1.1
الفهم المشترك لمخزون الثغرة



RAND *RR1751-1.1*

فقط فرد أو منظمة، وأخرى تعرفها مجموعة من الأفراد أو المنظمات. وقد تهدف بعض هذه الجهات الفاعلة إلى إبقاء الثغرات خاصة بدون إصلاح وقابلة للاستغلال (أي ثغرة يوم صفر)، في حين قد تهدف أخرى إلى سلوك مسار معاكس.[14] ونوزّع الثغرات وفق الفئات التالية:

1. الثغرات التي لا تكشف للعلن:
   a. تبقى محصورة بفرد واحد أو منظمة واحدة فقط
   b. يعرفها أكثر من فرد أو منظمة (والتي قد تُعتبر معادية أو صديقة)، حيث يرغب الطرفان في بقائها سرية
   c. يعرفها أكثر من فرد أو منظمة، حيث يهدف أحد الطرفين إلى جعل الثغرة علنية
   d. يعرفها البائع المتضرر (وربما آخرون بشكل سري) وقد تم إصلاحها لكن لم يتمّ الإفصاح عنها علنًا على أنها ثغرة أمنية[15]
2. الثغرات المعروفة للعامة والتي يمكن أن يعلم بها البائع و:
   a. لم يتمّ إصلاحها أو تعديلها
   b. تمّ إصلاحها مع تعديل جزئي فقط
   c. تم إصلاحها وتعديلها بالكامل

مع مرور الوقت، تصبح بعض الثغرات المعروفة في المجال الخاص علنية لعدة أسباب (اكتشفها وأفصح عنها باحث مستقل أو الشركة المتضررة أو مجموعة خاصة، الخ). وبالتالي، مع مرور الوقت، تتداخل مخزونات الثغرات الخاصة والعامة (كما يظهر في الشكل 1.2).

ويمكن تقسيم أولئك الذين يبحثون عن ثغرات ويطورون برمجيات إكسبلويت للاستخدام الخاص، في معظم الوقت، إلى مجموعتين متعارضتين: الفريق الأرزق والفريق الأحمر.[16] وقد تشمل المجموعتان وكالات حكومية ومتعاقدين في مجال الدفاع وباحثين عن الثغرات، يتألفون من أفراد مستقلين أو مجموعات أفراد ومتاجر تجارية لتطوير برمجيات إكسبلويت.

يُظهر الشكل 1.2 نظرة مبسّطة وبعض الأمثلة على العناصر التي يضمها كل مجال.

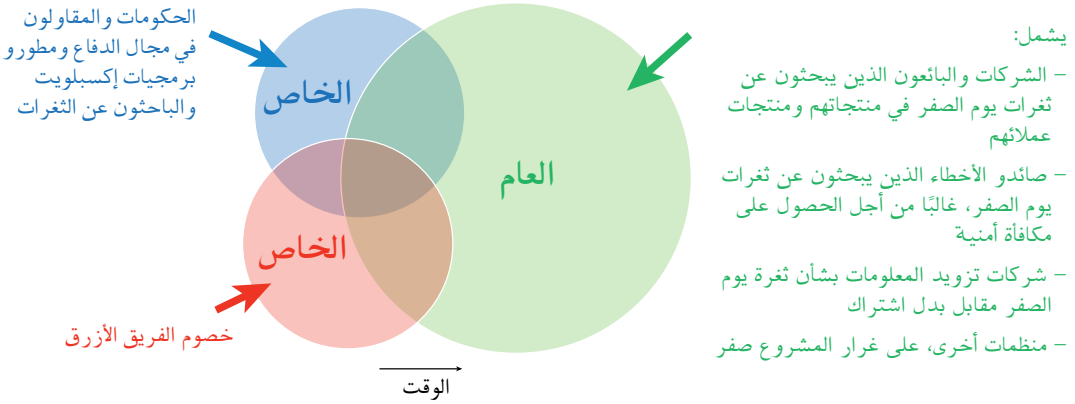
قد يساعد الإفصاح عن ثغرات يوم الصفر المعروفة للمجموعات الخاصة لكن ليس للعامة – الموجودة في التداخل البنفسجي في الشكل 1.2 – الفريق الأزرق على تعزيز دفاعاته لأنه سيتمكن من حماية نفسه ضد ثغرات يوم الصفر التي وجدها خصومه أيضًا (الفريق الأحمر). من ناحية أخرى، إن الإفصاح العلني عن ثغرات يوم الصفر الموجودة في القسم الأزرق المحض قد يلحق الضرر بالعمليات الهجومية التي يشنها الفريق الأزرق ويعود بالفائدة على الفريق

[14] المشروع صفر (Project Zero) التابع لغوغل (Google) خير دليل على ذلك.

[15] غالبًا ما يتمّ تجاهل هذه الفئة بشكل خاص في منشورات أخرى حول ثغرات البرمجيات.

[16] عمليًا، قد يتألف الفريق "الأحمر" من عدة مجموعات خاصة، لكن في هذه الحالة، نقوم بجمعها معًا لتمثّل أولئك الذين قد يستخدمون ثغرات يوم الصفر ضد مصالح الفريق الأزرق. فضلًا عن ذلك، ثمة فرضية بأنه إن كانت الثغرة خاصة، فهي (1) قيد الاستعمال و(2) تستخدمها دولة قومية أو مجرم. لكن باحثي الثغرات الذين تحدثنا إليهم أشاروا إلى وجود مجموعة من الثغرات يستهان بها ومنسية (أي تبقى خاصة) لأنها غير نافعة وربما تتمتع بخصائص تجعل عملية الإفصاح مع البائعين صعبة وتستغرق وقتًا طويلًا.

الشكل 1.2
نظرة مبسّطة على هوية من يشمله مخزون الثغرة

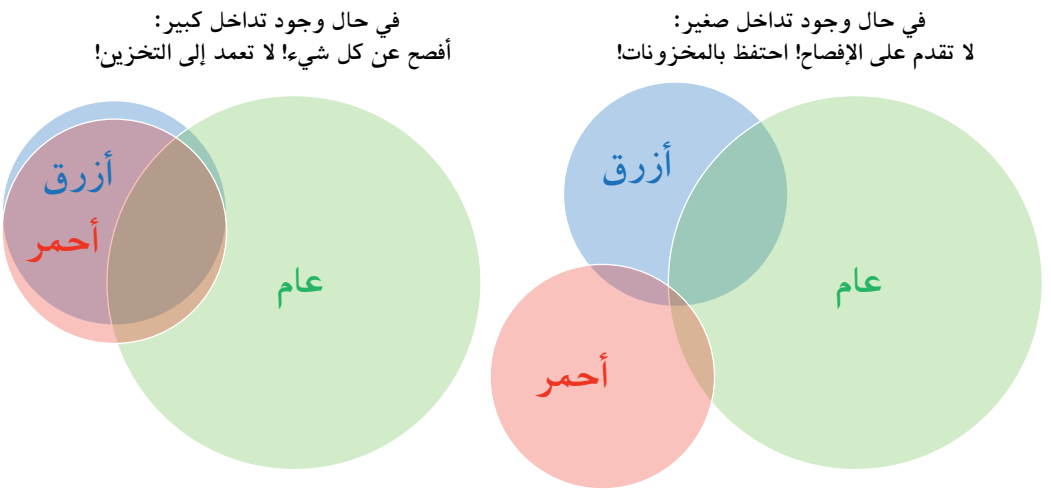


RAND *RR1751-1.2*

الأحمر لأن هذا الأخير يعلم بوجود ثغرات يوم الصفر التي اكتشفها الفريق الأزرق الذي يبقي مع ذلك ثغرات يوم الصفر الخاصة به لنفسه.

أما المجهول الأكبر فهو قدر التداخل الموجود بين الفريقين الأزرق والأحمر. ويُظهر الشكل 1.3 سيناريوهين متطرفين. من شأن تداخل كبير أن يدعم حجة الإفصاح عن جميع الثغرات المكتشفة وإطلاقها. لكن في حال كان التداخل صغيرًا نسبيًا، عندها تكون المجموعتان بصدد

الشكل 1.3
نظرة مبسّطة على قرارات الاحتفاظ



RAND *RR1751-1.3*

اكتشاف ثغرات مختلفة، والكشف عما يعرفه الفريق الأزرق قد يضعفه. في هذه الحالة، قد يكون الاحتفاظ بالثغرة قيّمًا.[17]

البيانات المستخدمة في هذا البحث

حصلت مؤسسة RAND على مجموعة من المعلومات حول برمجيات إكسبلويت لاستغلال ثغرات يوم الصفر من خلال علاقات بحثية.[18] إنها مجموعة بيانات غنية، حيث إن بعض برمجيات إكسبلويت اكتشفها آخرون وأخرى لا. وتغطي هذه المجموعة 14 عامًا (منذ 2002 وحتى 2016) وتحوي معلومات بشأن أكثر من 200 برمجية إكسبلويت لاستغلال ثغرات يوم الصفر والثغرات التي استفادت منها، أكثر من نصفها غير معلوم.[19] وقد بلغ العدد النهائي لبرمجيات إكسبلويت التي تضمنتها البيانات 207، بعد إزالة ما بين 20 و30 برمجية بسبب الحساسية التشغيلية.[20]

تغطي البيانات عدة جوانب من برمجية إكسبلويت وثغرة البرمجية التي تستفيد منها: أنواع المنتجات والبائعون وأنواع المصادر وأنواع الثغرات وأنواع برمجيات إكسبلويت. كما أنها تتضمن تواريخ تحدد متى اعتبرت ثغرة ما جديرة بالاستغلال ليصبح لها برمجية إكسبلويت فعالة بالكامل، ومتى تمّ استحداث برمجية إكسبلويت وما إذا كان طرف ثالث اكتشف الثغرة ومتى. وتتضمن أيضًا معلومات حول المحلل الذي صاغ برمجية إكسبلويت والنشرة الأمنية ذات الصلة (متى كان ذلك مطبقًا).

وقد احتوت مجموعة البيانات على عدة فئات لكل مدخلة، وثمة الكثير من الأشياء التي يمكن دراستها. نجري في هذا التقرير بعض التحاليل المهمة لتحديد مجموعة من المقاييس المرتبطة بحالة ثغرة يوم الصفر ومعدلات صمودها ومعدلات الاصطدام والتكاليف (المحددة بحسب الوقت الضروري لتطوير برمجيات إكسبلويت). يمكن إيجاد المزيد من التفاصيل في الملحق G "المزيد من المعلومات حول البيانات".

وقد أتت البيانات من مجموعة بحث عن الثغرات (نطلق عليها اسم BUSBY لحماية مجهوليتها).[21] وقد عمل بعض باحثي BUSBY لصالح دولة قومية (لذا فإن مستوى مهارتهم ومنهجيتهم تنافس تلك التي تتمتع بها فرق دولة قومية)، كما تستخدم الدول القومية العديد من منتجاتBUSBY. وبالتالي، تعتبر هذه البيانات بديلة عما يمتلكه الفريق الأزرق، أو مجموعة استخدام خاص متطورة، ويمكن استعمالها للنظر في عدة أسئلة ترتبط بثغرات يوم الصفر، بما في ذلك ديمومة ثغرات يوم الصفر وأمد حياتها وأنواع الثغرات التي تدوم لأطول فترة قبل أن يكتشفها الآخرون وأنواع المنتجات التي تضم ثغرات يوم الصفر القابلة للاستغلال.

[17] كما ذكر سابقًا، إن حساب هذا القرار ليس سهلًا.

[18] يتمتع مصدر البيانات بخبرة كبيرة في المشاركة في أسواق الاستخدام الخاص (رمادي) والمعرفة العامة (أبيض).

[19] اعتبارًا من تاريخ توقف بياناتنا (1 آذار (مارس) 2016)

[20] الرقم الدقيق غير معلوم.

[21] Busby هو نوع من الجيوش التي حظيت بالشعبية مطلع القرن العشرين. وقد وجدنا أنه مصطلح مستعار مناسب لأنه (1) مرجع مرح لعدة أنواع من القبعات (البيضاء، الرمادية والسوداء) بالنسبة إلى الباحثين عن الثغرات و(2) غالبًا ما تستخدم العناصر العسكرية برمجيات إكسبلويت لاستغلال ثغرات يوم الصفر.

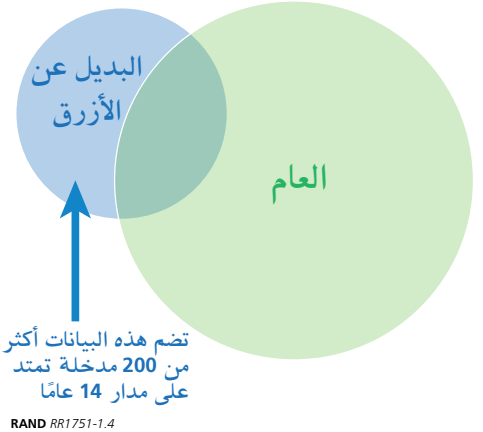
وبشكل مثالي، كنا نرغب في الحصول على بيانات مماثلة عن الفريق الأحمر (أي خصوم الفريق الأزرق، أو مجموعات أخرى تركز على الاستخدام الخاص)، لمراقبة التداخل بين الفرقين لكننا عجزنا عن ذلك. عوضًا عن ذلك، قمنا بالتركيز على التداخل بين الفريق الأزرق والعامة (أي القسم ذي اللون الأخضر الضارب للزرقة في الأشكال أعلاه) لاستنتاج ما قد يشكل خط الأساس لما يمتلكه الفريق الأحمر. ونقوم بذلك استنادًا إلى فرضية أن ما يجري في المجموعات العامة شبيه إلى حدّ ما بما يحدث ضمن مجموعات أخرى. ونقر بأنها فرضية ضعيفة نظرًا إلى الاختلاف المقبول المحتمل بين تركيبة المجموعات العامة والخاصة وتركيزها وحوافزها ومدى تطورها، لكنها البيانات الوحيدة المتوافرة حاليًا.

وقد أدركنا من خلال بياناتنا حجم التداخل بين ثغرات الاستخدام الخاص وثغرات المعرفة العامة (أي التداخل بين اللونين الأزرق والأخضر في الشكل 1.4).

ويركز بحثنا على خصائص الثغرات المتداخلة بين اللون الأزرق (أو أي بديل منطقي عن الأزرق) والعامة (أي القسمين الأزرق والأخضر الضارب للزرقة). وقد سعينا إلى فهم حجم التداخل وأمد حياة أو ديمومة ثغرات يوم الصفر التي تمّ الاحتفاظ بها قبل كشفها للعلن. ونستخدم هذا الأمر لإيجاد أساس لتداخل ثغرات يوم الصفر وأمد حياتها. ومن شأن هذه الحدود الدنيا توفير مؤشر والمساعدة على تأمين المعلومات اللازمة لإجراء المناقشة حول التخزين. وقد تؤدي الرؤى الإضافية إلى مخزون بعض خصوم الفريق الأزرق الأكثر تطورًا (أو المجموعات الأخرى التي تكتشف هجمات يوم صفر للاستخدام الخاص) إلى تقديرات أكثر دقة للتداخل وأمد الحياة.

وفي حين كانت نيتنا إجراء التحليل واستخلاص النتائج من البيانات والتدابير الكمية، سعينا أيضًا إلى الحصول على الخبرة النوعية من آخرين ينخرطون في مجتمع الأبحاث عن الثغرات. بالإضافة إلى الحوارات المعمقة مع الذين وفروا لنا البيانات، تحدثنا إلى باحثين عن الثغرات من ثلاث شركات من السوق الرمادي (أي الباحثين الذين يقومون بصياغة برمجيات إكسبلويت للحكومات أو العملاء الذين يستخدمونها لأغراض خاصة)، وإلى الذين هم في شركات

الشكل 1.4 بياناتنا



تجارية كبيرة ولديهم فرق بحثية خاصة عن الثغرات، وإلى الذين ينتمون إلى شركات تدير برامج مكافأة أمنية، وخبراء آخرين لديهم منشورات حول هذا الموضوع أو يتمتعون بخبرة من نوع آخر في مجال شراء ثغرات يوم الصفر أو بيعها أو الوساطة بشأنها أو تطويرها.

منهجية البحث وجمع البيانات

تتضمن بياناتنا خصائص عن كل برمجية إكسبلويت
لقد قمنا بإعداد مجموعة بياناتنا من خلال انتقاء المعلومات ذات الصلة من رمز برمجية إكسبلويت ذاتها. وجمعنا معلومات عن مختلف خصائص كل برمجية إكسبلويت:

- نوع الثغرة (عند مستوى عال ومنخفض)
- نوع المنصة (عند مستوى عال ومنخفض)
- فئة برمجية إكسبلويت
- نوع رمز المصدر
- البائع

تجدون في الملحق G "المزيد من المعلومات حول البيانات" مزيدًا من المعلومات إلى جانب تجزئة الترددات.

جمع البيانات وتحديد تواريخ أمد الحياة
أمضى ثلاثة محللين أسبوعًا يعملون بشكل منهجي من خلال مخازن رمز برمجية إكسبلويت ومستندات مفصلة لأكثر من 200 برمجية، حيث تمكنوا من استخلاص معلومات عن الثغرة المستغلة والمنصة المتضررة والحالة وأمد الحياة ومعلومات أخرى ذات صلة.
في ما يلي التواريخ الثلاثة المحتملة لكل برمجية إكسبلويت:

1. **تاريخ الميلاد:** تاريخ اكتشاف ثغرة واتخاذ القرار بأنها تستحق استحداث برمجية إكسبلويت فعالة بالكامل لاستغلالها.[22] وقد حصلنا على هذا التاريخ في معظم الأحيان بفضل تسجيل الدخول إلى المخزن. وعندما لم تكن تسجيلات الدخول إلى المخزن متاحة، استخدمنا تواريخ توثيق برمجية إكسبلويت عندما كان مطور البرمجية يدون الجدول الزمني الخاص بعمله(ها). وفي الحالات التي لم تكن فيها هذه التواريخ متوافرة، استخدمنا تاريخ إعداد المستند (أي تاريخ بدء مطور برمجية إكسبلويت بالكتابة للمرة الأولى عن الثغرة التي

[22] يمكن استغلال مجموعة فرعية فقط من ثغرات يوم الصفر، لذا عندما يكتشف مطورو برمجيات إكسبلويت الثغرات، عليهم اتخاذ القرار بشأن ما إذا كانوا قادرين على تطوير برمجية إكسبلويت. وقد تدخل العديد من العوامل في الحسبان عند اتخاذ قرار تطوير برمجية إكسبلويت لاستغلال الثغرة. على سبيل المثال، قد يكون هناك ثغرات أو أخطاء أخرى تمنع استغلال الثغرة المستهدفة (راجع النقاش حول الأخطاء المانعة" لاحقًا في هذا التقرير)، وقد لا تؤدي الثغرة إلى شيء مفيد (حتى الآن)، أو قد يكون لدى مطور برمجية إكسبلويت عدة برمجيات مشفرة بالفعل في المكان نفسه ولم يستخدمها، فيكرس حينها وقته للعمل على ثغرات أخرى (مثلًا في هذه المجموعة من البيانات، وصلت 14% فقط من المدخلات إلى مرحلة "إثبات المفهوم" ولم يتم استحداث برمجيات إكسبلويت فعالة بالكامل).

اكتشفها). في بعض الحالات، اعتمدنا على ذاكرة مطور برمجية إكسبلويت الذي كان يعمل على برمجية محددة. وفي حالات أخرى، عندما لم يكن تاريخ الاكتشاف الأولي متاحًا، اعتبرنا أن تاريخ الميلاد هو تاريخ تطوير إثبات مفهوم عابر لبرمجية إكسبلويت.[23]

2. **تاريخ النضوج:** تاريخ ظهور برمجية إكسبلويت فعالة بالكامل وموثوقة (أي برمجية إكسبلويت الجاهزة للتسليم إلى عميل أو لاستخدامها في العمليات).[24] وقد حصلنا على هذا التاريخ من توثيق برمجية إكسبلويت وسجلات المخزن.[25]
3. **تاريخ الموت (متى طبق ذلك):** تاريخ اكتشاف طرف ثالث خارجي للثغرة والإفصاح عنها للعامة. ينطبق ذلك فقط على بعض الثغرات وبرمجيات إكسبلويت في مجموعة بياناتنا.[26] وقد استخدمنا محركات البحث لإجراء أبحاث عبر الإنترنت عن السلاسل والمكتبات والوظائف والمتغيرات وغيرها من معلومات التعريف الفريدة التي كانت تتعلق بالثغرة أو بأساس برمجية إكسبلويت أو بأداة تحقيق الاستغلال. إلى جانب استخدام أساليب محرك البحث الأساسية، قمنا بالتدقيق في مختلف المخازن حيث يتمّ الإبلاغ عن الثغرات المعلومة (مثلًا، أنظمة الثغرات الشائعة والانكشاف (CVE) وقاعدة بيانات الثغرات الوطنية (NVD) وقاعدة بيانات ثغرات مفتوحة المصدر (OSVDB) وباغتراك (Bugtraq) وبوابة سيكيوريتي فوكس (SecurityFocus)).[27] وقد أمضى المحللون الثلاثة بين 30 و45 دقيقة تقريبًا في البحث عن كل ثغرة. وفي الحالات التي لم تظهر فيها أي نتائج، بحثنا عن أي ثغرة معلنة للمنتج وحللنا النتائج لمعرفة ما إذا كانت الثغرة المحددة مطابقة أو ما إذا توفر دليل منطقي يدعونا إلى افتراض مطابقتها.[28] أحيانًا، كان اكتشاف ثغرة مكتشفة أساسًا يعني إزالة القوائم الإلكترونية.[29] في حالات محدودة، وصف الناس الثغرة الموجودة في قائمة

[23] ما يعني الرمز الذي يؤكد أو يثبت أن ثغرة ما قابلة للاستغلال. للمزيد من التفاصيل حول إثبات مفهوم برمجيات إكسبلويت، راجع الفصل الثاني، الملحق A، "دورة تطوير برمجيات إكسبلويت".

[24] ثمة بعض الحالات التي كان من الضروري فيها متابعة العمل لجعل برمجية إكسبلويت موثوقة لعدة نسخ، لكن لم يتمّ تدوين الجدول الزمني.

[25] تعتبر هذه التواريخ دقيقة إلى حدّ ما، إذ يكون مطورو برمجيات إكسبلويت متحمسين وفخورين وتواقين إلى مشاركتها مع زملائهم عندما يستحدثون برمجية إسكبلويت فعالة بالكامل وموثوقة.

[26] تتمثل طريقة أخرى لتحديد ما إن كانت ثغرة يوم صفر وبرمجية إكسبلويت لا تزال فعالة في تجربة البرمجية على النسخة الأحدث من المنتج. لكننا طبقنا ذلك على بضع حالات، وفي معظم الحالات لم نحصل على الموافقة لاختبار برمجية إكسبلويت لكي نعرف إن كانت لا تزال حية. وبالتالي، اعتمدنا المنهجية المذكورة هنا لتحديد إن كانت الثغرة لا تزال ثغرة يوم صفر أو لا.

[27] ندرك أن هذه المخازن لا تشمل جميع الثغرات المكشوفة علنًا. وقد قدر بعض الخبراء في مجال الإفصاح عن الثغرات الذين تحدثنا إليهم أن الثغرات الواردة في قاعدة بيانات ثغرات مفتوحة المصدر (OSVDB) (التي تسمى الآن VulnDB) تمثل بين 80 و90% من جميع الثغرات المعلومة (بما فيها جميع الثغرات الواردة في أنظمة الثغرات الشائعة والانكشاف (CVE)، قاعدة بيانات الثغرات الوطنية (NVD))، وأن الثغرات الواردة في CVE/ NVD تمثل نحو 50% من الثغرات الموجودة في OSVDB/VulnDB. أما الثغرات التي لا تشملها قاعدة بيانات OSVDB (وبالتالي ليست ضمن قاعدتي (CVE/NVD)) كانت أخطاء كبيرة حسبما يُعتقد في المواقع الإلكترونية الحية – وكانت غالبًا محور تركيز برامج المكافأة الأمنية. وقد أشاروا إلى أنه، بالنسبة إلى الثغرات الموجودة ضمن OSVDB/VulnDB وليس ضمن CVE/ NVD، ثمة احتمال كبير بعدم وجود وسيلة إصلاح أو أن ماسحات الثغرات لم تجدها بعد. ويُعتقد أن الثغرات موجودة ضمن أنظمة مرتبطة بالضبط الصناعي والسيارات والطب.

[28] سمحت لنا هذه الخطوة باستخلاص نتائج مهمة. على سبيل المثال، استخدم BUSBY بشكل خاص مدخلة ضمن نظام CVE صنفت علنًا ثغرة بأنها تحجب الخدمة لتصعيد المزايا (وهو أمر غالبًا ما يُعتبر أكثر خطورة وتعقيدًا من حجب الخدمة).

[29] ذكر البعض أن مناقشة ثغرة ضمن قائمة إلكترونية متاحة للعلن لا تضمن اعتبار الثغرة "معلومة" أو "ميتة" أو أنها لم تعد ثغرة يوم صفر. نحن ندرك أنه من المقبول اعتبارها حية وثغرة يوم صفر، لكن لأغراض بحثنا، اعتبرناها ميتة.

إلكترونية على أنها مجرد خطأ (وليست ثغرة أمنية)، وما من دليل على أنه تمّ إصدار نشرة أمنية أو نشرة عامة للثغرة. ولا نزال نعتبر هذه الثغرات ميتة، بما أنه جرى الكشف عن المعلومات، بغض النظر عما إذا كانت صُنفت قضية أمنية. وفي حالات قليلة جدًا، انتاب مطوري برمجية إكسبلويت الذين تعاون معهم المألفون "شعور داخلي" بأنه جرى اكتشاف ثغرة، لكننا لم نتمكن من إيجاد أي إثبات.[30] في هذه الحالات، اعتبرنا أن برمجية إكسبلويت "غير مؤكدة".

لتحديد التواريخ، استخدمنا الطوابع الزمنية في سجلات المخزن. وبما أن بعض برمجيات إكسبلويت كانت أقدم وكان مطور برمجية إكسبلويت الفعلي هو من قام بتوثيقها، كانت بعض المستندات أكثر دقة من غيرها. وفي وقت كانت فيه تفاصيل برمجية إكسبلويت دقيقة دائمًا، لم يكن الحصول على التواريخ المحددة ممكنًا دائمًا. نتيجةً لذلك، نفتقر إلى معلومات عن تاريخ الميلاد والنضوج أو لا نعرف سوى الشهر والسنة المرتبطين بمجموعة فرعية من برمجيات إكسبلويت البالغ عددها 207.[31] وقد أثر ذلك على الحجم النموذجي الخاص بالتحليل، إذ إنه عنى أنه كان بإمكاننا استخدام برمجيات إكسبلويت التي نملك جميع المعلومات بشأنها فقط لإتمام تحليل الصمود وحساب أمد الحياة المتوقع. وفي الحالات التي كنا نملك فيها الطابع الزمني الخاص بالشهر والسنة لتاريخ الميلاد فقط، عجزنا عن التحديد مطلع الشهر، لذا في بعض الحالات، قد يصل المدى الزمني إلى 30 يومًا. لكن بما أن الأمر كان محصورًا بتاريخ الميلاد، وليس بتاريخ النضوج، قد تكون الأرقام متحفظة نسبةً إلى الوقت الضروري لتطوير برمجية إكسبلويت. وفي حالات قليلة، اعتمدنا على الذاكرة الشخصية للذين كانوا منخرطين مباشرة في تطوير برمجية إكسبلويت لتحديد الوقت الضروري للاستغلال.[32]

لقد استخدمنا تواريخ الميلاد والموت لحساب معدلي الديمومة والاصطدام، باعتبارها تواريخ تحدد اكتشاف الباحثين الخاصين للمرة الأولى (الميلاد) والإفصاح للعامة للمرة الأولى (الموت). كما أن تاريخ النضوج مهم في حساب الوقت اللازم لتطوير برمجية إكسبلويت فعالة بالكامل.

تنظيم هذا التقرير

تمّ تنظيم ما تبقى من التقرير على الشكل التالي. يوفر الفصل الثاني خلفية إضافية والمزيد من التعمق في طبيعة ثغرات يوم الصفر وعملية تطوير برمجية إكسبلويت ونماذج الأعمال والباحثين الأمنيين أنفسهم. وقد يرغب القراء الذين لديهم إلمام بهذا الموضوع في الانتقال مباشرة إلى الفصل الثالث، حيث نغوص في تحليل البيانات ونناقش مختلف مراحل حياة

[30] لتحديد وجود ثغرة بشكل دقيق، اضطررنا إلى تنزيل النسخة القابلة للتطبيق من رمز المصدر والبحث عن الثغرة.

[31] استجابت 61% فقط من بياناتنا للمعايير التي احتجنا إليها لإجراء هذه العمليات الحسابية.

[32] نحن ندرك أن الأشخاص يميلون إلى عدم الدقة عندما يتعلق الأمر بالذاكرة. وعليه، فالذين كانوا يحاولون التذكر نجحوا في ذلك لعدد صغير من العينة التي نعتمدها، وكانوا خبراء في هذا المجال، لذلك ربما كانت ذاكرتهم أكثر دقة.

برمجيات إكسبلويت، وأرجحية الصمود وتقديرات أمد الحياة المتوقع، ومعدل الاصطدام، ونلقي نظرة أولية على تكلفة تطوير برمجية إكسبلويت. أما الفصل الرابع فهو بمثابة خاتمة تشمل التداعيات والتوصيات. وبالنسبة إلى الملحقات، فهي تضم معلومات إضافية عن دورة تطوير برمجيات إكسبلويت (الملحق A) والباحثين عن الثغرات (الملحق B)، وتشمل دراسة حالة عن كيفية تأثير عمليات التخفيف على قابلية الاستغلال (الملحق C)، وتحقيق في اصطدامات قريبة (الملحق D)، ونظرة مهمة على التكاليف والاعتبارات السعرية لبرمجية إكسبلويت لاستغلال ثغرات يوم الصفر (المحلق E)، ورسوم بيانية إضافية (الملحق F) ومعلومات عن البيانات (المحلق G) التي قد تهم القارئ. أما الملحق H فيوفر مسردًا بالمصطلحات والتعريفات لتعزيز فهم القارئ لثغرات البرمجيات وبرمجيات إكسبلويت.

نقاش إضافي حول ثغرات يوم الصفر

قدّم الفصل الأول خلفية موجزة عن ثغرات يوم الصفر. ويقدم هذا الفصل المزيد من التفاصيل حول الطبيعة المعقدة لثغرات يوم الصفر وعملية تطوير برمجيات إكسبلويت ونماذج الأعمال والباحثين الأمنيين. ومتى أمكن، نقدم أمثلة ملموسة من بياناتنا ونقاشاتنا، فضلًا عن المزيد من المعلومات في الملحقات.

طبيعة ثغرات يوم الصفر

الثغرات ديناميكية

الثغرات ديناميكية، وقد تختلف فائدة خطأ ما بحسب مكانه في الرمز.[1] علاوةً على ذلك، لا تعتبر جميع الأخطاء ثغرات ولا يمكن استغلال جميع الثغرات بشكل مفيد. وقد تتغير فائدة الخطأ في الرمز بين يوم وآخر. فثغرة غير قابلة للاستغلال في المقام الأول قد تصبح في نهاية المطاف قابلة للاستغلال في قواعد رمز مستقبلية (والعكس صحيح).[2] ويتمّ باستمرار تطوير نماذج وتقنيات نفاذ جديدة تسمح باستخدام ثغرات لم تكن قابلة للاستغلال، كما يجري اعتماد عمليات التخفيف بطريقة مماثلة، ما يمنع أي استعمال آخر للثغرات القابلة حاليًا للاستغلال.[3] في إحدى الحالات، اكتشف باحثو BUSBY ثغرة شائبة في التصميم سمحت بتنفيذ رمزي لثنائيات اعتباطية عن بعد في موقع محدد، ما يعتبره الكثيرون ثغرة عالية الأثر، لكن كان يجب جمعها مع ثغرات أخرى لإتاحة النفاذ الكتابي إلى هذا الموقع وتحقيق أي أثر (أي، إن عجز الشخص عن

[1] على سبيل المثال، يكون خطأ في الإسناد المؤشري الباطل ضمن مجال المستخدم صعب الاستغلال عمومًا، فتاريخيًا كان استغلال خطأ مماثل في النواة أسهل. أما الثغرة في إسناد مؤشري باطل فتبرز عندما يتمّ مسح عنوان الذاكرة أو جعله صفرًا قبل حصوله على مرجعه الأول. وضمن مجال المستخدم، على الفرد أن يعمل بواسطة مخصصات ديناميكية ونسخ مجمعة مختلف، إلخ، لكن في النواة، يوجد عادة توزيع مجمع واحد فقط.

[2] على سبيل المثال، قد تحقق ثغرة قابلية استغلال حديثة العهد بسبب تغييرات في الرمز أو الجامع الذي أزال "الخطأ المانع" – الأخطاء التي لا تسفر عن شيء مفيد بل تسبب عطلًا أو تعيق أخطاء أخرى يمكن استغلالها. وقد فكر بعض مطوري برمجيات إكسبلويت في تقديم الأخطاء المانعة إلى البائع المتضرر ليتمّ إصلاحها لكي يصلوا إلى الخطأ القابل للاستغلال. بعبارة أخرى، بدل أن يكون مطور برمجية إكسبلويت قادرًا على إدخال ثغرة في الرمز المصدر (لأنها لا تشكل جزءًا من فريق التطوير الأساسي للبائع)، يسمح "الخطأ المانع" المقدم للإصلاح بإدخالها.

[3] على سبيل المثال، إدخال رسوم تدفق التواصل في ويندوز 10.

كتابة رمز يجب تنفيذه، لا يكتسب امتلاك القدرة على التنفيذ القيمة نفسها). وقد يكون انتظار تحول خطأ ذي أثر كبير إلى ثغرة قابلة للاستغلال مثيرًا للإحباط، فقد لا تظهر أي وسائل لجعل الخطأ قابلًا للاستغلال أبدًا، أو قد لا يتمّ اكتشاف الثغرة وإصلاحها قبل التمكن من استخدامها.[4] وعليه، في حين قد يستغرق التحقق من قابلية الاستغلال وقتًا طويلًا بعد إطلاق كل رمز جديد، يرى العديد من مطوري برمجيات إكسبلويت أن تعقب الثغرات التي سبق أن اكتشفوها لكن لا يجري استغلالها حاليًا أقل تكلفة من محاولة اكتشاف ثغرات جديدة تتطلب عادةً مصادر أكثر من تطوير برمجيات إكسبلويت.[5] ونظرًا إلى أن خطورة الثغرة أو تأثيرها قد يختلف بحسب موقعها في الرمز أو الاعتماد على ثغرات أخرى أو هدف المشغل الذي يستخدم سبيل استغلال، قد يكون وضع درجة ثابتة مضللًا.[6]

أسس تطوير برمجيات إكسبلويت والاعتبارات المتعلقة بها

يُعتبر تطوير برمجية إكسبلويت موثوقة مليئًا بالتحديات، فقد تؤدي الاعتبارات التشغيلية (بما فيها المشغل وعمليات التخفيف) إلى إثارة الشكوك وتراجع الموثوقية. وبشكل مثالي، تعمل برمجية إكسبلويت على كل تهيئة ونسخة للهدف، لكن الاختبار الحقيقي الوحيد يتمثل بتشغيل برمجية إكسبلويت على الهدف المنشود. وبما أن إجراء الاختبار ضمن سياق تشغيلي غير ممكن عمومًا، يتخذ الباحثون عن الثغرات مقاربات مختلفة لتقليص الشكوك وزيادة الموثوقية في السياقات التشغيلية. فيستحدث بعضهم برامج جداول تُظهر مختلف النسخ والتهيئات التي استخدمت في الاختبار، ويختبرون كل ترتيب ألف مرة تقريبًا. في حين يقوم آخرون بالتدقيق في برمجية إكسبلويت، والسياق التشغيلي للهدف إلى أقصى حد ممكن، ويحاولون اكتشاف ما لا يزالون يجهلونه واحتمالات الفشل. وهذا يعني ضمان أن تكون العناصر الأساسية (وحدات بناء برمجية إكسبلويت) موثوقة قدر المستطاع، ما قد يعتمد إلى حدّ كبير على نوع الثغرة التي يتمّ استغلالها.[7] وبما أن برمجية إكسبلويت قد تتضمن عدة ثغرات مرتبطة ببعضها البعض، يمكن جمع

[4] على سبيل المثال، اعتُبرت ثغرة في الإسناد المؤشري الباطل وجدها الباحثون في مجال المستخدم غير قابلة للنفاذ؛ فما كان من الباحثين سوى انتظار كشف نسخة جديدة من الرمز عن ثغرة في النواة، ما وفر سبيلًا للولوج إلى الثغرة الموجودة في مجال المستخدم واستحداث برمجية إكسبلويت مفيدة. في حالة أخرى، أخبرنا الباحثون عن ثغرة في تطبيق يستخدم عمومًا بشكل لاسلكي كانت لتتطلب الكثير من الجهود للاستغلال (مثلًا قد تتطلب معدات خاصة للنفاذ). وبالتالي، وضعها المحللون جانبًا وعملوا عليها مجددًا عدة مرات للتحقق من بروز بعض الثغرات الجديدة التي قد تساعد في الاستغلال.

[5] في الحالات التي تضمنتها بياناتنا حيث لم يتم تطوير برمجية إكسبلويت (رغم اكتشاف ثغرات قابلة للاستغلال)، لم يجر استغلال حوالى نصفها لأن الأمر استلزم بعض القدرات أو تحقق ظروف معينة، ولم يتم استغلال الأخرى لأن المحللين كانوا يتمتعون أساسًا بتغطية كبيرة في قاعدة الرمز المحددة.

[6] عليه، قد تكون تصنيفات الأثر، على غرار تلك الموجودة في نظام تصنيف الثغرات المشتركة (CVSS) ، مفيدة من الناحية الدفاعية، ويُعتقد أنه يمكن تطبيقها في الوقت والسياق المحددين حيث تم اكتشاف الثغرة. من ناحية أخرى، قد لا تكون تصنيفات الأثر والخطورة دائمًا دقيقة: على سبيل المثال، وجد باحثو BUSBY ثغرة كانت بمثابة ثغرة سوء إدارة ذاكرة ذات تأثير كبير واستغلوها (ما يسمح بكتابة خارج الحدود)، لكن اعتُبرت ضمن أنظمة الثغرات الشائعة والانكشاف أنها تشكل عقدة متوسطة كخطأ في قطع الخدمة.

[7] على سبيل المثال، الثغرات الموجودة في أماكن محددة من الرمز (مثلًا مجال المستخدم) أو التي تنتمي إلى أنواع معينة (مثلًا هجمات command injection، SQLi وفائض التكديس) موثوقة عمومًا. لكن الثغرات التي تؤدي إلى فائض في التكويم أو تتطلب وقتًا أو حالات تسابق قد لا تكون موثوقة بالقدر نفسه.

ثغرة موثوقة عادةً بثغرة غير محددة لاستحداث برمجية إكسبلويت غير موثوقة، أو قد يختلف سلوك برمجية إكسبلويت تعمل بشكل موثوق على نظام مادي إلى حدّ كبير على جهاز افتراضي.[8]

قد يحاول الباحثون أيضًا تحديد المعلومات الأساسية المتعلقة بالهدف، على غرار حمولة النظام وتهيئته، وتأخر الشبكة، والتجزئة والهندسة، والاستعمال، وهندسة الذاكرة وتوافرها. كما لا بد من الأخذ في الحسبان التدابير المضادة وعوامل التخفيف، على غرار أدوات موازنة الحمولة وآليات الرصد، ومنع التحديث التلقائي، ووجود سبيل للإصلاح (سواء كان ذلك مطبقًا أو لا)، وأنظمة كشف التسلل ومنعه. وقد يكون من المفيد أن يعلم مطوّر برمجية إكسبلويت طبيعة مغروس المتابعة المستهدف.[9] إن الشكوك حيال موثوقية برمجية إكسبلويت تعتمد أيضًا على مهارات المشغل الذي يستخدم برمجية إكسبلويت هذه ومدى معرفته بالبيئة المستهدفة.[10] وبغض النظر عن المقاربة المعتمدة، لا تزال الاختبارات تجرى في المختبرات مع تباينات طفيفة من حيث إعدادات تصحيح الأخطاء والمجموعات اللغوية والبطء والنطاق الترددي التي قد تؤثر على نجاح برمجية إكسبلويت وموثوقيتها في سياق تشغيلي.

قد تكون برمجية إكسبلويت سبيل استغلال "لمرة واحدة" يسمح بمحاولة واحدة للاستغلال (مثلًا فائض تكديس واحد في عملية سيصبح غير متوافر في حال فشل برمجية إكسبلويت)، أو قد يتطلب استغلال عدة أشياء بشكل متسلسل أو تكرار محاول الاستغلال عدة مرات.[11] وما إن يُقدم الباحثون عن الثغرات على بيع برمجية إكسبلويت أو تأجيرها إلى عميل، تكون العملية قد أنجزت، ويعود قرار كيفية استعمال البرمجية وما إذا كان يتوجب استعمالها إلى العميل.

دورة تطوير برمجيات إكسبلويت

تتألف عملية تطوير برمجية إكسبلويت من عدة خطوات، وقد تسلك كل خطوة عدة مسارات. فالمرحلة الأولى من تطوير برمجية إكسبلويت تتمثل في استحداث إثبات المفهوم، وذلك من أجل إثبات أنه يمكن تحقيق تنفيذ الرمز لحالة أو نسخة واحدة. فبرمجية إكسبلويت فعالة

[8] نتحدث هنا عن موثوقية برمجيات إكسبلويت، وليس المغروسات أو "الحمولات". فالموثوقية ضرورية لبرمجيات إكسبلويت ولكنها ليست مضمونة دائمًا. غير أن موثوقية المغروسات أساسية. وفي حين توفر برمجية إكسبلويت ولوجًا أوليًا، يحافظ المغروس على هذا الولوج ويضمن بقاءها ويخلف بعض الآثار. وإن كان المغروس غير موثوق، فقد يؤدي إلى بروز خانة الشاشة الزرقاء وإلحاق الضرر بالعمليات الحساسة. من الصعب للغاية الحفاظ على المغروسات، لذا فإن أسعارها مرتفعة، حيث يقدر ثمن المغروس بما يصل إلى مليون دولار.

[9] مثلًا، إن كان العميل يخطط لاستخدام حمولة لإخراج البيانات، عندها يجب أن يتنبه مطورو برمجيات إكسبلويت إلى خطة الخروج والتدابير المضادة (مثلًا جدران الحماية) القائمة، إن وجدت.

[10] يعود قرار الاستعمال التشغيلي إلى العميل. يمكن لمطور برمجية إكسبلويت أن يوفّر الملاحظات فقط ويأمل أن يدرك المشغلون ما الذي يجب فعله. في حال لم تعمل برمجية إكسبلويت ضمن السياق التشغيلي كما حصل في البيئة الاختبارية، قد يتعين على المشغل تحديد خصائص البيئة لمستهدفة التي تؤثر على برمجية إكسبلويت (مثلًا استعمال برمجية إكسبلويت خلال حركة الإنترنت الكثيفة، وبالتالي خلال نطاق ترددي بطيء ومكتظ إلى حدد كبير، في نظام مستهدف. قد يدرك مشغل متمرس ضرورة الانتظار إلى ما بعد أوقات العمل لاستعمال برمجية إكسبلويت.). ويتفوق بعض المشغلين على غيرهم في هذا المجال.

[11] أحيانًا، يمكن استخدام عدة برمجيات إكسبلويت في مراحل مختلفة للحد من الشكوك وزيادة الموثوقية. على سبيل المثال، يمكن استخدام الثغرات التي تسرب المعلومات في المراحل الأولية للتعرف على بصمات والحصول عليها، من أجل المساعدة تطوير برمجية إكسبلويت للمتابعة. في حال تطلّبت الحمولة المستخدمة من قبل المشغل بعد الاستغلال الأولي الخروج، عندها قد يكون من الضروري معرفة خطة الخروج (مثلًا ما هي جدران الحماية أو التدابير المضادة التي قد تُستخدم، أو ضمن أي بروتوكول يجب أن تنحصر الحمولة). وقد يكون جزء آخر من برمجية إكسبلويت مسؤولًا عن إلغاء وجود البرمجية من خلال إزالة السجلات، ضمان الاستمرار (أي عدم وجود أعطال) والديمومة والسلاسل القائمة وإعادة حالة النظام المستهدف إلى ما كانت عليه.

بالكامل قادرة على توسيع إثبات المفهوم ليكون مفيدًا لكل نسخة محتملة من النظام.[12] وغالبًا ما تترافق برمجيات إكسبلويت مع مستندات للمشغل (تشمل معلومات حول بطء النظام ومتطلبات النطاق الترددي والآثار الثانوية المحتملة على غرار مدخلات السجلات المتروكة وما الذي يجب إلغاؤه). للمزيد من التفاصيل حول دورة تطوير برمجيات إكسبلويت وبعض العوامل التي تدخل ضمن هذه العملية، يرجى مراجعة الملحق A "دورة تطوير برمجية إكسبلويت".

الأشخاص في نطاق ثغرات يوم الصفر

يركز بعض الباحثين عن الثغرات على إيجاد ثغرات يوم الصفر وتقديمها إلى البائعين المتضررين مقابل دفع تكلفة في بعض الأحيان وفي أحيان أخرى عرفانًا بالجميل.[13] وغالبًا ما تدخل هذه الثغرات في المجال العام من خلال تقارير وشعارات منشورة عن الثغرات.[14] وتتنامى هذه المجموعة من صائدي الأخطاء "أصحاب القبعات البيضاء" (white hat) بسبب ازدياد شعبية برامج المكافأة الأمنية المقدمة من الشركات، على غرار هاكروان (HackerOne) وباغ كراود (BugCrowd). ويمتلك العديد من بائعي البرمجيات مجموعاتهم الداخلية الخاصة المكلفة بإيجاد ثغرات يوم الصفر في برمجيات الشركة وأي برمجية تستخدمها الشركة.[15] وتوفر شركات مثل إكزودس إنتليجنس (Exodus Intelligence) وزي دي آي (ZDI) وآي ديفانس (iDefense) "موجز ثغرة يوم صفر" يمكن للمشتركين لديهم استخدامه للاختبار الدفاعي ولتطبيق تدابير وقائية في المنتجات. وعمومًا تعود الثغرات إلى البائع المتضرر. ويكون الهدف غالبًا 90 يومًا، رغم أن هذا الإطار الزمني قد يتغير.[16] يركز الباحثون الذين يشاركون في برامج المكافأة الأمنية أو يبيعون المعلومات عمومًا على إيجاد ثغرة وتطوير برمجية إكسبلويت أساسية لإثبات المفهوم خاصة بها. كما يهدف الباحثون ضمن المشروع صفر (Project Zero) في غوغل (Google) إلى إيجاد الثغرات وتزويد البائع بالنتائج التي توصلوا إليها في غضون 90 يومًا، لكنهم لا يتوقفون عن حدود إثبات المفهوم، بل يسعون إلى استحداث برمجية إكسبلويت فعالة بالكامل بغية تحديد الأثر الحقيقي. وعليه، فهم يشبهون الباحثين الساعين إلى إيجاد ثغرات للاستعمال الخاص.

يتمّ أحيانًا إصلاح الثغرات التي يحتفظ بها بشكل سري أو تبقى ببساطة بدون إصلاح.[17] فالمجموعة التي تبحث عن ثغرات يوم الصفر للاستعمال الخاص تتألف عمومًا من باحثين يسعون إلى اكتشاف ثغرات ليستخدموها لأغراض تشغيلية، سواء دفاعية أو هجومية، وقد تضم جهات فاعلة من الدولة القومية من "أصحاب القبعات الرمادية" (gray hat) أو مقاولين في مجال الدفاع

[12] في هذه المرحلة تكتسب برمجية إكسبلويت مبررًا إضافيًا: مثلًا، تسرب ذاكرة أو معلومات أو خوارزمية القوة القاسية أو قائمة محتملة من العناوين لتجربتها على نسخ مختلفة.

[13] تندرج هذه المجموعة في الجانب الأيمن من الشكل 1.1.

[14] هناك عدة قواعد بيانات متاحة للعامة تتضمن معلومات بشأن الثغرات المكتشفة (التي يمكن إصلاحها أو لا). وكان أشهرها قاعدة NVD و ODSVDB المتاحة مجانًا، التي أصبحت الآن قاعدة VulnDB المدفوعة (شركة الأمن القائمة على المخاطر Risk-Based Security، 2017).

[15] مثلًا، مجموعة تالوس (Talos) من سيسكو (Cisco).

[16] عند اكتشاف بعض ثغرات يوم الصفر التي بقيت مجهولة لأربع سنوات، غيرت إكزودس إنتليجنس (Exodus Intelligence) سياستها القائمة على عدم الإفصاح مطلقًا عن أي ثغرة إلى الإفصاح عنها بعد 90 يومًا من إخطار البائع (إكزودس إنتليجنس، 2016).

[17] تقع هذه المجموعة في الجانب الأيسر من الشكل 1.1.

(من أي بلد)، أو مجرمي الإنترنت من أصحاب "القبعات السوداء" (black hat) أو من هواة لديهم دوافع مختلفة.[18] ولا يتمثل الهدف الأكثر تعقيدًا في إيجاد ثغرات يوم الصفر فقط بل في إيجاد برمجيات إكسبلويت فعالة بالكامل لها.

وفي حين أن الأرقام المحددة مجهولة، يقدر الكثيرون أن عدد الأشخاص ضمن المجموعة "العامة" يفوق من حيث المقدار الأسي على الأقل عدد الأشخاص الذين يبحثون عن ثغرات يوم الصفر ويستحدثون برمجيات إكسبلويت للاستعمال الخاص. وهناك ثلاث فئات من الباحثين عن الثغرات بشكل أساسي.

فأولئك الذين يصنفون في الفئة الأولى يتمتعون بمهارات كبيرة ويتفاوتون بين مقاول منفرد يعمل بمفرده، وفريق من عشرات الأشخاص، وصولًا إلى شركة تضم (بضع) مئات. وتعتبر برمجيات إكسبلويت التي يضعها أعضاء هذه الفئة ذات جودة وموثوقية عالية. كما أن معظم مكونات هذه الفئة تكون من الأسواق الرمادية، أي تسعى لإيجاد برمجيات إكسبلويت للاستعمال الخاص أو لبيعها إلى الحكومات أو المقاولين في مجال الدفاع، رغم أن باحثي الأسواق البيضاء الذين يجدون ثغرات ويعمدون إلى مشاركتها فيما يمثل جزءًا من المعرفة العامة موجودون أيضًا، وقد تبرز بعض السيولة بين القبعات الرمادية والبيضاء رهنًا بالتمويل والموارد المتاحة (مثلًا تأتى الكثير من الباحثين في المشروع صفر التباع لغوغل من فرق حكومية وفرق السوق الرمادي). ويُعتقد أن عدد الباحثين في هذا المستوى المرتفع يتراوح بين 300 و3 آلاف.[19] هذا ويعتقد الكثيرون أن صياغة سبيل استغلال موثوق تزداد صعوبة لأن البائعين باتوا يتمتعون بخبرة أكبر في مجال الإصلاح ولأن عمليات التخفيف تهدف إلى تغطية فئات واسعة من الثغرات.

أما الذين يتواجدون في الفئة المتوسطة، فلديهم القدرة على الكشف عن الثغرات وصياغة برمجيات إكسبلويت للثغرات والأهداف المنخفضة المستوى، لكنهم قد يعجزون عن صياغة برمجيات إكسبلويت في مواجهة أهداف صعبة وقد يعتمدون على أدوات قديمة أو على إعادة تحديث رمز برمجية إكسبلويت قائمة لاستعمالهم الخاص. ويعتقد البعض أن هذه الفئة المتوسطة هي التي تقود أسواق الجرائم الإلكترونية.

وفي القاع هناك صائدو الأخطاء الذين يقفون عند حدّ اكتشاف الثغرات وربما إظهار برمجية إكسبلويت افتراضية. وبالنسبة إلى المهارة المطلوبة عند هذا المستوى، فيمكن تعليمها. ويتفق معظم الخبراء على أن عدد الباحثين في الفئة الدنيا تفوق من حيث المقدار الأسي على الأقل عدد أولئك الذين ينتمون إلى الفئة العليا. مثلًا، تضم منصة باغ كراود (BugCrowd) أكثر من 26 ألف باحث (رغم أن عدد صائدي الأخطاء الناشطين ليس واضحًا، حيث أن عشرة منهم فقط شكلوا 23% من إجمالي المدفوعات)، وقد تردد أن بضعة آلاف منهم فقط اختاروا اصطياد الأخطاء في منصة هاكروان (HackerOne)، وحوالى 8 آلاف ساهموا في برنامج الكشف عن ثغرة "وويون"

[18] من الأمثلة على ذلك، وجد قرصان حاسوب يُدعى "فينيس فيشر" (Phineas Fisher) ثغرة يوم صفر واستعملها لاختراق أنظمة مجموعة السوق الرمادي "هاكينغ تيم" (Hacking Team) لاستخراج البيانات ورمز المصدر (زورابيديان Zorabedian، 2016).

[19] قدر أحد الخبراء وجود 300 باحث في خدمة الولايات المتحدة و1500 حول العالم. في حين قدر آخرون وجود ما يتراوح بين ألف وألفي باحث كحد أقصى عالميًا. من جهته، توقع شخص آخر ملم بهذا المجال وجود 3 ألف باحث يعملون لصالح مقاولي الدفاع الأمريكيين مع عدد مماثل يعمل لصالح دول أخرى. مثلًا، يُعتقد أن شركة تينسينت كين سيكيوريتي لاب (Tencent's Keen Security Lab) الصينية تضم نحو 3 آلاف باحث أمني، رغم أنه من المستبعد أن يتمتعوا جميعهم بأعلى المهارات والقدرات.

(Wooyun).[20] ويعتقد أن أنواع الثغرات الموجودة في كل فئة مختلفة نسبيًا، رغم احتمال التداخل. وتجدون تفاصيل إضافية عن الخصائص، طول السيرة المهنية وخبرة مطوري برمجيات إكسبلويت في الملحق ب " الباحثون عن الثغرات: من الذي يبحث عن الثغرات؟"

نماذج الأعمال

أسواق ثغرات يوم الصفر

تنامت أسواق ثغرات يوم الصفر خلال السنوات القلية الماضية وتتميز من حيث هوية الشاري الأولي وطبيعة الثغرة العامة مقابل الخاصة ووجهة استعمال الثغرة المقررة.[21] أما أعضاء السوق الأبيض، فيسعون فورًا إلى نقل الثغرات إلى البائع المتضرر (حيث يعمدون غالبًا إلى نقلها إلى مجال المعرفة العامة) واستخدامها لأغراض دفاعية. وفي السوق الرمادي (الذي يسمى أيضًا السوق الحكومي)، تبقى الثغرات خاصة، وتستخدم لأغراض هجومية أو دفاعية، ويمكن الإفصاح عنها في نهاية المطاف إلى البائع المتضرر، رغم أن الأمر ليس مضمونًا لأنها تباع عادةً أولاً إلى حكومة أو جيش أو مقاول في مجال الدفاع.[22] كما أن الأسواق السوداء تبيع ثغرات يوم الصفر للاستخدام الجنائي أو لأغراض غير قانونية وتسعى إلى إبقاء الثغرات خاصة.

ويجني الذين يطورون برمجيات إكسبلويت فعالة بالكامل، ثم يبيعونها أو يؤجرونها أموالاً أكثر من الذين يكتشفون الثغرات أو ينشئون إثبات المفهوم، لأن شاري الثغرات يشترون ضمانة على أن برمجية إكسبلويت موثوقة وفعالة، وفي بعض الحالات يمكن تجنب رصدها. وعليه، تُعتبر برمجيات إكسبلويت لإثبات المفهوم عادةً هدفًا نهائيًا لباحثي القبعات البيضاء، حيث تعمل الأسواق الرمادية والسوداء بواسطة برمجيات إكسبلويت فعالة بالكامل.[23]

مع مرور الوقت تبدل الشارون والبائعون. فخلال السنوات الماضية، كان من الشائع استعمال الأطراف الثالثة والوسطاء و"القواطع" (cutouts) لنقل برمجيات إكسبلويت، لكن الحال اختلف اليوم عمومًا. فبعد مرور ما يكفي من الوقت ومع توفر الروابط الصحيحة، تفضل منظمات تطوير برمجيات إكسبلويت العمل مباشرة مع العميل لأن ذلك يتيح تحكمًا أكبر بمصير برمجية إكسبلويت. وقد أشارت إحدى فرق بحوث الثغرات التي تحدثنا إليها إلى أن تجربتها كانت سيئة مع الوسطاء الذين يعاودون بيع برمجيات إكسبلويت إلى عدة أطراف، وفي حالة أخرى، يقوم الوسيط بمشاركة برمجية إكسبلويت مع شركائه الخاصين. علاوةً على ذلك، بسبب تقلص حجم

[20] إن أكثر من نصف الباحثين الذين تسجلوا في موقع باغ كراود هم من الهند والولايات المتحدة (باغ كراود، 2016؛ جاو Zhao وغروسكلاغس Grossklags وليو Liu، 2015).

[21] يمكن استخدام الإطار الزمني بوصفه وسيلة أخرى للتمييز بين الأسواق: في السوق الأبيض، تُمنح الثغرة غالبًا إلى البائع من دون تأخير. وفي السوق الرمادي، قد تعود الثغرة إلى البائع بعد تأخير أو بعد استخدامها من شخص آخر. أما في السوق الأسود، فلا تعود الثغرة بشكل عام إلى البائع أبدًا.

[22] يعتقد البعض ضمن السوق الرمادي أنهم جزء من السوق الأبيض لأنهم يعملون ضمن إطار القانون، ويستحقون بالتالي اللون "الأبيض". غير أننا نبني الفارق على المستلم الأساسي للثغرة أو برمجية إكسبلويت الخاصة بالثغرة، وليس على مبدأ إن كان المشاركون في السوق يعملون ضمن إطار القانون أو خارجه. وبما أن العميل في السوق الرمادي يميل إلى أن يكون جزءًا من الحكومة أو مرتبطًا بها، يصح استخدام مصلح السوق الحكومي بديلاً عن السوق الرمادي.

[23] انظر الفصل الرابع من ليبيكي وأبلون وويب (2015) للمزيد من المعلومات والفوارق بين الأسواق الثلاثة.

السوق، يقوض استخدام وسيط من طرف ثالث هوامش الأرباح الدنيا. في هذا السياق، أخبرنا وسيط سابق أن "مهنة الوسيط لا تدر الأموال ولا مستقبل لها". ويتمثل اتجاه آخر في أن الشارين والبائعين في السوق الرمادي ينتقلون إلى السوق الأبيض.[24]

اختلاف نماذج الأعمال

برز تطوير برمجيات إكسبلويت لقاء ربح منذ أواخر التسعينيات ومطلع العقد الأول من الألفية الثالثة. ومنذ ذلك الحين، تبدلت نماذج الأعمال والتركيز، وبات تطوير برمجيات إكسبلويت واستغلال ثغرات يوم الصفر يقوم على عملية ذات طابع أكثر رسمية. فعلى سبيل المثال، في الأيام الأولى، بحث مطورو برمجيات إكسبلويت عن ثغرات قابلة للاستغلال في كل شيء، وكانت برمجيات إكسبلويت تميل إلى أن تكون خاصة بنظام التشغيل، في حين توقف تطوير برمجيات إكسبلويت عند مستوى إثبات المفهوم. والآن، يركز مطورو برمجيات إكسبلويت على العميل أكثر، بينما تميل برمجيات إكسبلويت إلى أن تكون خاصة بالعملية والتطبيق إلى حدّ أكبر، ويطلب الشارون أن تكون برمجيات إكسبلويت فعالة بالكامل وتعمل على عدة نسخ وتهيئات.

هذا ويقدم بعض اللاعبين في السوق الرمادي على تأجير برمجيات إكسبلويت أو استئجارها أو التعاقد بشأنها أو منح تراخيص لها لكي يحتفظوا بالملكية الفكرية. وتكون مدة عقد الإيجار (حصري عادة) حوالي سنتين. وعند انقضاء أجل التعاقد، يمتلك العميل غالبًا خيار إعادة تأجير برمجية إكسبلويت. ويحصل كل عميل على ترخيص محدد لبرمجية إكسبلويت يحدد ما يحق له فعله وما يحظر عليه (رغم أنه ما من وسيلة أمام الشركة أو مطور برمجية إكسبلويت لتطبيق هذا). ويختار بعض البائعين الجهات التي يريدون بيع منتجاتهم لها، لذا قد يكون لديهم مجموعة محدودة من العملاء.[25]

وتصر مجموعات أخرى تعنى بالبحث عن الثغرات على أن الفائدة الوحيدة بالنسبة إليهم تكمن في البيع، لأن العديد من العملاء يتشاركون المعلومات أساسًا ويصعب تطبيق القيود المفروضة على التراخيص.[26] فضلًا عن ذلك، بما أن عدد العملاء ضئيل، تتراجع حماسة المجموعة التي تعنى بالبحث عن الثغرات لتطبيق برمجية إكسبلويت ستستخدم في مكان واحد فقد.

بشكل عام، يطلب الشارون الحصرية، على الأرجح لمتطلبات أمنية تشغيلية. وأحيانًا يتمّ بيع أو تأجير برمجيات إكسبلويت المتشابهة (من حيث فئة الخطأ أو المكتبة المستهدفة أو تقنية برمجية إكسبلويت المستخدمة) إلى عدة عملاء، لكن عمومًا فقط في حال تمكن البائع من إثبات أن اكتشاف برمجية واحدة لن يؤدي إلى اكتشاف آخر. ولدى بعض الشارين قواعد داخلية تحد عدد الثغرات التي يمكنهم الاحتفاظ بها في النظام، لذا قد يدفعون المال لمطور برمجيات إكسبلويت لإبقاء البرمجية "على الرف" كي يتم شراؤها أو استئجارها في وقت لاحق.

---

[24] عزا أحد الخبراء هذا إلى أن السوق الأبيض هو الوحيد القابل للبيع حاليًا لأن البيئة التسويقية ووجهة النظر تبدلت. في الماضي، كان على المرء أن يضع نفسه في مكانة "سيد الفنون القاتمة" لجذب العملاء. وما إن كان باحث عن ثغرات يكتسب هذه الميزة، كان بإمكانه الانتقال إلى السوق الأبيض.

[25] كشفت مجموعة واحدة تعنى بالبحث عن الثغرات أنها تبيع برمجيات إكسبلويت إلى 3 أو 4 جهات، في حين أشارت مجموعات أخرى إلى رقم مماثل.

[26] غير أن هؤلاء البائعين قد يمتلكون "رمزًا مساعدًا" يمكن إعادة استخدامه لدعم العملاء خلال اختبار برمجية إكسبلويت إلى أن يصبح العميل قادرًا على صياغة رمز إنتاج خاص به للاستعمال التشغيلي. يشار إلى أن البائع يحتفظ بالرمز المساعد.

يحدد بعض من يشترون برمجيات إكسبلويت ما يريدون من استخدامهم للبرمجية (جزء معين من برمجية في تهيئة محددة) في حين قد يرغب بعضهم في الحصول على البرمجية لهدف معين، حيث يقولون للمطورين "أريد دخول هذا النظام المستهدف أو هذه المنظمة المستهدفة بأي ثمن، ولا تهمني الطريقة". أحيانًا، يحدد الشارون آليات برمجية إكسبلويت، وما إذا كانوا يقبلون بحدوث أخطاء أو لا، وأي منصات أو هندسيات يجب دعمها (مثلًا 32-بت مقابل 64-بت)، وما إذا كان لا بد لبرمجية إكسبلويت من أن تكون هجومًا وسيطًا (man-in-the-middle exploit) أو إن كانت هناك متطلبات وقت معينة.[27] بشكل مثالي، يتشارك العملاء ما يريدون من الباحث عن الثغرة التركيز عليه بسبب حساسية الأهداف والتشغيل، ما يزيد العائد على الاستثمار بالنسبة إلى مطور برمجية إكسبلويت والشاري على السواء: فالأول لديه شيء يركز عليه وضمان أكبر للشراء، في حين يحصل الأخير على ما يريد. في هذه الحالات، قد يقدم الباحث عن الثغرة مجموعة برمجيات إكسبلويت متوافرة ويترك للعميل قرار الاختيار. وبما أن مطوري برمجيات إكسبلويت لا يدركون ما يخبئه المستقبل، يجدون أنفسهم أحيانًا أمام برمجيات غير مباعة تكون بمثابة "طبقة تخزين مؤقت" (caching layer) إلى أن يطلب عميل ما برمجية إكسبلويت محددة. يطرح ذلك خطر إفساد البرمجية أو خطر اكتشاف ثغراتها. وقد قدرت مجموعة تعنى بتطوير برمجيات إكسبلويت عدم بيع 50% من برمجياتها المطورة.

غالبًا ما يزيد مطورو برمجيات إكسبلويت (وشركات تطوير برمجيات إكسبلويت) دخلهم من خلال تقديم استشارات وبيع مؤشرات الثغرات

رغم الادعاءات السائدة أنه يمكن جني مبالغ كبيرة من المال من خلال استحداث برمجيات إكسبلويت لاستغلال ثغرات يوم الصفر،[28] يركز عدد قليل من الشركات والمنظمات اليوم حصرًا على تطوير برمجيات إكسبلويت.[29] فهي تكمل عادة تطوير برمجيات إكسبلويت الخاصة بها بأعمال أخرى على غرار تقديم خدمات اختبار نسبة الانتشار أو مؤشرات الثغرات، أو بيع

[27] يشير الهجوم الوسيط (man-in-the-middle) أو الآلة الوسيطة (machine-in-the-middle) إلى مهاجم قائم بين مرسل ومتلق شرعيين ينقل وربما يبدل التواصل بين الطرفين.

[28] على سبيل المثال، قدر فيدلر (Fidler) وغرانيك (Granick) وكرينشاو (Crenshaw) (2014) بلوغ حجم السوق الرمادي 4 ملايين دولار إلى 10 ملايين. وقد أبلغتنا مجموعات تعنى بالبحث عن الثغرات أن الحكومة الأمريكية تستثمر مبالغ أكبر بكثير على تقييم المنتجات والاشتراك في مؤشرات الثغرات، وأقل على شراء برمجيات إكسبلويت وثغرات يوم الصفر.

[29] قدرت مجموعة تعنى بالبحث عن الثغرات تراوح عائدات تطوير سبيل استغلال بين 6 و7% من إجمالي الأعمال، حيث يتأتى الجزء الأكبر من العائدات من استشارية تجارية وأدوات بيع. ويكمن اعتبار هذه الوظائف المؤقتة الاستشارية التجارية بمثابة "اصطياد خطأ قصير الأمد". وفي معظم الأحيان، خلال تقييم أمني، سيتم اكتشاف ثغرة يوم صفر في موقع العميل، وعليه سيصار إلى الإفصاح عنها فورًا. في هذا الإطار، أشارت إحدى الشركات البحثية في مجال الثغرات إلى أن ذراع الوظائف المؤقتة الاستشارية الخاصة بها وجدت عددًا أكبر من ثغرات يوم الصفر من ذراعها البحثية. وقد يعزى ذلك إلى واقع أنه، لكي تتمكن شركة بحثية من اكتشاف ثغرة يوم صفر، عليها إعداد البرمجيات، قاعدة بيانات موجودة على الخادم، وبنية تحتية قبل البدء بأي عملية تدقيق؛ علمًا بأن عملية الإعداد هذه قد تستغرق شهرًا. غير أن المستشار يملك أساسًا بنية تحتية قائمة كاملة في موقع العميل، لذا يكون أحيانًا من الأسهل إيجاد ثغرة يوم صفر هناك. وتميل هذه هجمات يوم الصفر إلى أن تكون من نوع محدد – غالبًا نسخ لشركات (مقابل نسخ للاستعمال الشخصي أو لاستعمال العميل) أو برمجيات تعمل على خط رأس محدد (مثلًا برمجيات خاصة بمكتب محاماة أو منشاة رعاية صحية).

البيانات وأدوات الدعم والمنتجات.[30] يصعب قياس تطوير برمجيات إكسبلويت، وباستثناء المقاولين الأفراد الذين يعملون بمفردهم، ما من محفزات مالية في معظم الأحيان.[31]

ويعتقد أن حوالى عشرين شركة تبيع هذه المنتجات إلى الولايات المتحدة أو إلى جهات حليفة لها، في وقت يتراجع فيه هذا العدد باطراد.[32] وقد عزا بعض الأشخاص الملمين بالأسواق البيضاء والرمادية تراجع الأعداد إلى التكاليف الباهظة، زاعمين أن العديد من هذه الشركات وجدت أن نموذج برمجية إكسبلويت ليس مربحًا بقدر نموذج الخدمة أو أنه لا يحظى بقبول عام على غرار المكافآت الأمنية واصطياد الأخطاء في السوق العام.[33]

[30] يعتبر مؤشر الثغرات نموذجًا قائمًا على الاشتراك، حيث يحصل المشتركون على معلومات منتظمة حول ثغرات ثغرة يوم صفر لم تكتشف بعد. وغالبًا ما يستخدم المشتركون المؤشر لاختبار منتجاتهم مقابل ثغرات يوم الصفر بغية تأمينها بشكل أكبر، أو يستخدمون المؤشر خلال اختبارات الانتشار أو الالتزامات الاستشارية لاختبار منتجات عملائهم وشبكاتهم وأنظمتهم..

[31] يتمثل أحد الأسباب الرئيسية خلف تقاضي مطوري برمجيات إكسبلويت مبالغ طائلة في: إن شركة تضم سبعة أو ثمانية باحثين عن ثغرات من الفئة الأولى ويعملون بدوام كامل يتقاضون "مبلغًا من ستة أرقام يتراوح بين النطاق المتوسط إلى المرتفع" قد يدرون بين مليون و2.5 مليون دولار سنويًا من بيع برمجيات إكسبلويت، لكن حصة الأسد من هذه العائدات تعود إلى المحللين. وقد أبلغتنا إحدى الشركات أن عام 2015 سجل عائدات سلبية، لكنها واصلت تطوير برمجيات إكسبلويت لأنها "تحب هذا العمل" الذي يوفر لها الرضا تعتبره أهم من التعويض المادي.

[32] تتألف هذه الشركات من شركات صغيرة (تضم مجموعة محدودة من الباحثين) وأذرع شركات مقاولة تقليدية كبيرة مستقلة تعمل في مجال الدفاع (تضم عددًا كبيرًا من الباحثين). وكان أحد مطوري برمجيات إكسبلويت على دراية شخصية بوجود أكثر من 18 شركة مماثلة تبيع المنتجات إلى الولايات المتحدة. وكان يدرك أيضًا وجود عدد ضئيل من الشركات الأجنبية الأخرى التي تركز على حكوماتها. وعليه، تعتبر العديد من هذه الشركات جديدة في هذا المجال، وتأسست منذ بضع سنوات فقط.

[33] تعتبر شركة إكزودس (Exodus) مثلًا على إقدام شركة على تغيير نموذج أعمالها بشكل علني، حيث أنها أعلنت في شباط (فبراير) 2016 أنها ستبقي الثغرات لمدة 90 يومًا فقط للحصول على ملاحظات عملائها، ثم ستفصح عنها إلى البائع المتضرر لإصلاحها (إكزودس إنتليجنس Exodus Intelligence، 2016).

تحليل البيانات

كما ذكرنا في الفصل الأول، يجب أن يعلم أي شخص يريد اتخاذ قرار بشأن تخزين ثغرة ما أو الإفصاح عنها ما يلي:

1. **الحالة:** هل الثغرة هي ثغرة يوم صفر حقًا؟ هل هي "حية" (مجهولة للعامة) أم "ميتة" (معلومة للآخرين)؟
2. **الديمومة:** إلى متى ستبقى الثغرة غير مكتشفة وغير معلنة؟
3. **معدل الاصطدام:** ما هو احتمال أن يكتشف آخرون الثغرة ويفصحون عنها؟

قد يكون من المفيد أيضًا معرفة:

4. **التكلفة:** ما هي تكلفة تطوير برمجية إكسبلويت لاستغلال الثغرة؟

تتطرق الأقسام التالية إلى إجابات عن هذه الأسئلة، لا سيما ما هي الثغرات المجهولة أو المعلومة (الحالة)، وكم تدوم (أرجحية الصمود وأمد الحياة المتوقع)، ومعدل الاصطدام (التداخل). كما نتناول بشكل موجز بعض الاعتبارات الخاصة بالتكلفة لتطوير برمجية إكسبلويت. وبما أننا نستخدم منهجيات ومقاربات مختلفة لكل قسم، نناقش الأساليب والتحليل ضمن كل قسم.

وتضم مجموعة البيانات معلومات عن برمجيات إكسبلويت، لذا عندما نناقش الحالة وأمد الحياة المتوقع ومعدل التداخل والاصطدام، نشير إلى الثغرة أو الثغرات التي تستفيد منها برمجية إكسبلويت. هذا وتعني برمجية إكسبلويت في طور الاندثار (الموت) اكتشاف إحدى الثغرات التي كانت مجهولة في السابق (في حال وجود عدة ثغرات).[1]

[1] قد تضم سبيل إصلاح أيضًا، لكن عند المستوى الأساسي، نعرف "الموت" على أنه الاكتشاف من قبل الآخرين.

1. الحالة: هل الثغرة هي ثغرة يوم صفر حقًا؟ هل هي "حية" (مجهولة للعامة) أو "ميتة" (معلومة للآخرين)؟

تتمثل الممارسة الشائعة في تصنيف الثغرة ببساطة على أنها حية (مجهولة للعامة) أو ميتة (معلومة للآخرين)، غير أن تحليلنا أظهر أن **ثغرة ما تكون أكثر نقاوة سواء كانت حية أو ميتة.** ويُذكر أن هناك عدة فئات فرعية لكل منها، ما قد يجعل تصنيف ثغرة على أنها حية أو ميتة مضللًا ومبسطًا للغاية.[2] في ما يلي، نعدد مختلف حالات الثغرات وبرمجيات إكسبلويت ونفصل كيفية تصنيفها في قاعدة بياناتنا.

الحية: هناك طرق مختلفة لتكون بها الثغرة حية (مجهولة للعامة)

تُعتبر بعض الثغرات مجهولة للعامة ولا يزال المدافعون يبحثون عنها (لأنها تكن في النسخ الحديثة من النظام التشغيلي أو تستخدمها نسبة كبيرة من الناس). وتصنف في بياناتنا على أنها "**حية**" (n = 66).

هناك ثغرات تبقى في المنتج على الدوام لأن البائع لم يعد يصون الرمز أو يصدر التحديثات. ونصنف هذا النوع من الثغرات على أنها "**خالدة**" (n = 13). هذا لا يعني بالضرورة أن الثغرة تتمتع بخصائص معينة جعلتها خالدة: فهي قد تكون خالدة لأنها اكتُشفت قبل توقف البائع عن صيانة قاعدة الرموز وربما اكتُشفت لأن الوقت كان كافيًا.

الميتة: هناك طرق مختلفة لتكون بها الثغرة ميتة (معلومة للآخرين)

يمكن للباحثين الذين وجدوا الثغرات أو لطرف ثالث الإفصاح عن الثغرات المعلومة.

عندما يقوم باحث أصلي عن الثغرات بالكشف عن ثغرة، نصنفها على أنها "**قتلها BUSBY**" (n = 8). وقد يفعل الباحث ذلك لمساعدة مجتمع أمن المعلومات ككل من خلال مشاركة المعلومات أو للحصول على مقابل مادي لقاء اكتشاف والإفصاح عن ثغرة قد يشعر الباحث أنه يمكن لشخص آخر اكتشافها بسهولة وقريبًا.

وعندما يفصح طرف ثالث عن ثغرة، يترافق ذلك غالبًا مع نشرة أمنية وإصلاح، ونصنفها على أنها "**إصلاح أمني**" (n = 69) (Security Patch). وتتضمن النشرة الأمنية معلومات عن الثغرة، وقد تشمل أنظمة الثغرات الشائعة والانكشاف التي تصف الثغرة ودرجات نظام تصنيف الثغرات المشتركة التي من شأنها تفصيل الخطورة. وما يثير الاهتمام هو أن أنظمة الثغرات الشائعة والانكشاف لا توفر دائمًا معلومات دقيقة وكاملة عن خطورة الثغرة.[3]

---

[2] ندرك أن استخدام عبارة "ميتة" قد يكون مضللًا، لأن معظم التحركات الإلكترونية تستخدم الثغرات "الميتة" هذه من ثغرات الأخطاء المعلومة. وقد اقترح أحد الخبراء الذين تحدثنا إليهم علينا استخدام مصطلح "ثغرة غير موصوفة سابقًا في المنشورات"، لاسيما نظرًا إلى أن ردود الفعل العاطفية القوية إزاء كلمتي حية وميتة (ومسلحة) قد تصعب الحفاظ على الموضوعية والبقاء على الحياد في هذا المجال.

[3] يسهل فهم السبب الذي يدفع البائعين إلى عدم الكشف عن الكثير من التفاصيل الخاصة بالثغرات التي تمّ إصلاحها. انظر ريسكورلا (Rescorla) 2005. فبالنسبة إلى برمجية إكسبلويت ماتت ثغراتها من خلال إصلاح أمني في مجموعة بياناتنا، يشير نظام الثغرات الشائعة والانكشاف (CVE) إلى أنه قد تستخدم برمجية إكسبلويت قيمة معينة لإحداث تدفق قائم على التكديس، لكنه يعجز عن الإظهار أن برمجية إكسبلويت مصنعة بشكل مختلف قد تغير في الواقع قيمة عملية الإطلاق. في حالة أخرى، يشير نظام الثغرات الشائعة والانكشاف إلى الثغرة على أنها شيء يتسبب بقطع الخدمة عندما يتمّ استغلاله، لكن الباحثين عن الثغرات في BUSBY وجدوا أنها قد تقدم ميزة التصعيد، وهو أمر أكثر خطورة.

وأحيانًا، يتم اكتشاف ثغرة والإفصاح عنها في نظام واحد ولكنها تبقى مجهولة للعامة، ويمكن استغلالها بشكل خاص، في نظام آخر.[4]

لكن ليس لكل الثغرات التي يكتشفها الآخرون نشرة أمنية أو إصلاح. يقوم المطورون أو الباحثون عن الثغرات أحيانًا بنشر الخطأ أو الثغرة التي يجدونها في قائمة بريد إلكتروني أو مدونة إلكترونية أو كتاب. قد يكون الناشر على علم أو لا بأن الخطأ الذي تتم مناقشته هو في الواقع ثغرة أمنية، لذا ليس هناك نشرة أمنية مرتبطة بالثغرة. ونصنف هذه البيانات على أنها "**مشاركة علنًا**" (n = 6).

غير ذلك: تصنف بعض الثغرات ضمن فئات أخرى

تعتبر الثغرات التي يمكن إزالتها عبر مراجعات الرمز بدون الكشف عنها أو الإفصاح عنها ثغرات أمنية من النوع المثير للاهتمام. ونصنفها في خانة "**عامل إعادة هيكلة الرمز**" (n = 21). ويمكن استغلال هذه الثغرات في النسخ الأقدم، لتكون شبه حية (مثل الزومبي). لكن قد يعتبر البعض هذه الثغرات الزومبي "ميتة" لأنها لم تعد حية في أحدث نسخة للبرمجيات.[5]

لقد قمنا بتصنيف حالة بعض برمجيات إكسبلويت في مجموعة بياناتنا على أنها "**غير مؤكدة**" (n = 24). وتتألف هذه الخانة من برمجيات إكسبلويت بقيت حالة ثغراتها "غير مؤكدة" حتى بعد بحوث مكثفة عبر الإنترنت. وفي وقت الصياغة، كانت هذه الثغرات لا تزال في طور المراجعة لحالتها (أي كان باحثو BUSBY يتحققون من فعاليتها مع أحدث النسخ).

وتشمل مجموعة بياناتنا ثغرات حية (مجهولة للعامة) في قاعدة رموز واحدة ولكنها ميتة (معلومة للآخرين) في أخرى. فعلى سبيل المثال، جرى نقل رمز يضم ثغرة تسمح بتسريبات في آلية تحديد الوصول ضمن منتج مايكروسوفت إلى منتج ليس من صنع مايكروسوفت، فقد جرى اكتشاف الثغرة وإصلاحها في منتج مايكروسوفت لكنها تبقى غير مكتشفة (وغير مصلحة) في المنتج الآخر الذي لم تنتجه مايكروسوفت.

لقد جرى استحداث برمجيات إكسبلويت لكليهما وسجلناها مع الحالة المناسبة في مجموعة بياناتنا.

ويظهر الشكل 3.1 تقسيم البيانات بحسب الحالة، حيث يضم 3 فئات "حية"، "ميتة" و"غيرها"، في حين يظهر الشكل 3.2 تجزئة إضافية تشمل جميع الفئات. ويوفر الجدول 3.1 وصفًا موجزًا لكل حالة.

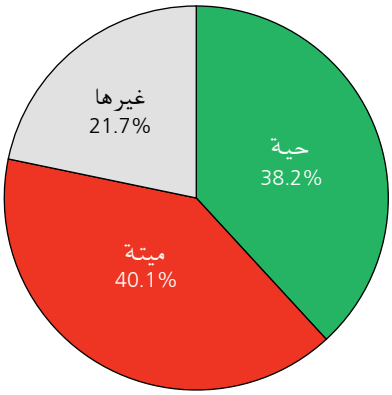
---

[4] على سبيل المثال، تضمنت إحدى برمجيات إكسبلويت التابعة لـBUSBY ثغرة وجدت في نسخة 32 بت من نظام معين لكن لم تكن قابلة للاستغلال. وفور صدور نسخة 64 بت، باتت الثغرة قابلة للاستغلال. وقد استمرت هذه الثغرة المعينة لأكثر من 5 سنوات قبل إطلاق إصلاح أمني.

[5] يمكن أحيانًا لعامل إعادة هيكلة الرمز تحويل قاعدة رموز سابقة إلى منتج جديد بالكامل، سواء لصالح مطوري برمجيات إكسبلويت أو لا. قد يغير عامل إعادة هيكلة الرمز بدون قصد قاعدة رموز فلا تعود الثغرات موجودة أو قابلة للاستغلال. في أحيان أخرى، يفتح عامل إعادة هيكلة الرمز المسارات أمام ثغرات لم تكن قابلة للاستغلال سابقًا. فمنتج كان نادرًا في السابق قد يصبح كثيفًا مع الثغرات القابلة للاستغلال، والعكس صحيح. على سبيل المثال، درست شركة رابيد7 (Rapid7) في 2012 قواعد الرموز القديمة الخاصة بأجهزة شبكة عاملة وفق بروتوكول دعم التركيب والتشغيل العالمي (Universal Plug and Play) التي تمّت إعادة هيكلتها في 2010، وأظهرت كيف كانت النسخ السابقة لهذه الأجهزة مليئة بالثغرات وكيف بقيت كذلك. وبما أن العديد من الأنظمة تحتفظ بالنسخ القديمة، بسبب غياب القدرة أو الموارد أو الوعي للتحديث، لا تزال العديد من موجهات لينكسيس (LinkSys) حول العالم ضعيفة. (انظر ما كتبه أتش دي مور hdmoore، 2013). وما يمكن استنتاجه هو أنه من المفيد العودة إلى نسخ الرمز القديمة والتدقيق بها، لا سيما (من وجهة نظر هجومية) إن كان معلومًا أنه من المستبعد تحديث هدف أو قطاع محدد.

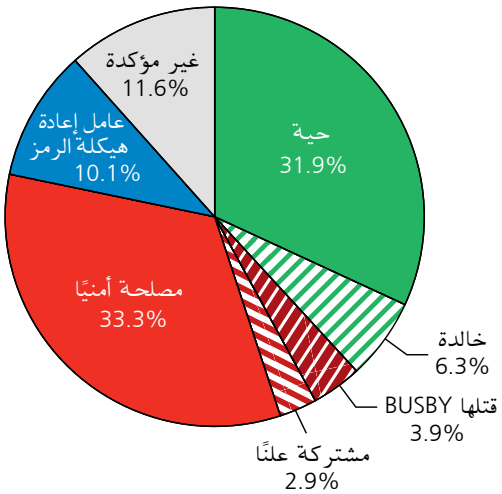
يفصل الشكلان 3.3 و3.4 الثغرات التي وجدها باحثو الثغرات في BUSBY بين 2002 و2015. هذا وتمّ إغفال بعض البيانات (بين 20 و30 برمجية إكسبلويت) خلال السنوات اللاحقة (فضلًا عن إغفال كافة البيانات المكتشفة في 2016) بسبب حساسة العمليات الراهنة.

الشكل 3.1
تجزئة بسيطة لحالة بياناتنا (n = 207)



RAND *RR1751-3.1*

الشكل 3.2
تجزئة إضافية لحالة بياناتنا (n = 207)

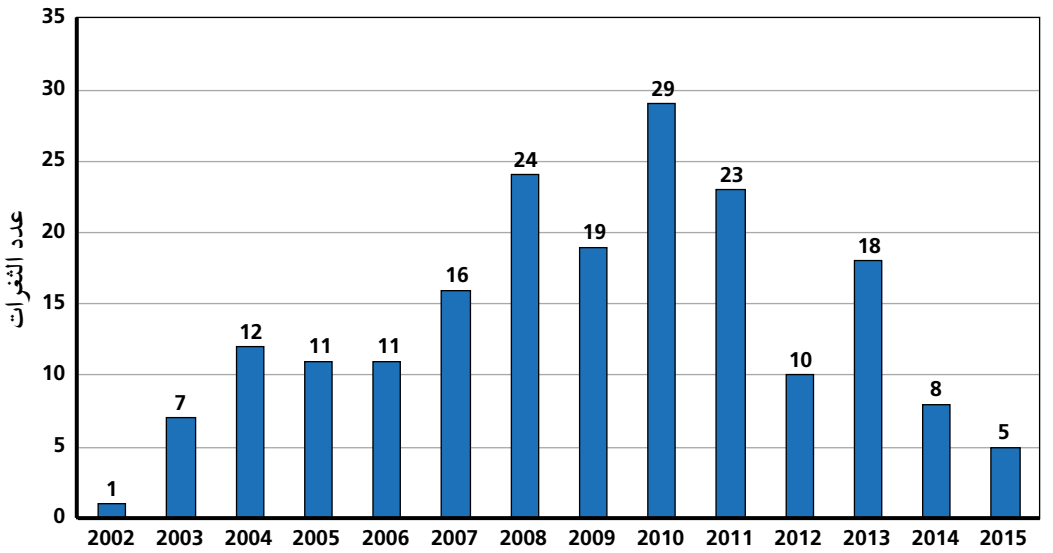


RAND *RR1751-3.2*

الجدول 3.1
وصف مختلف الحالات

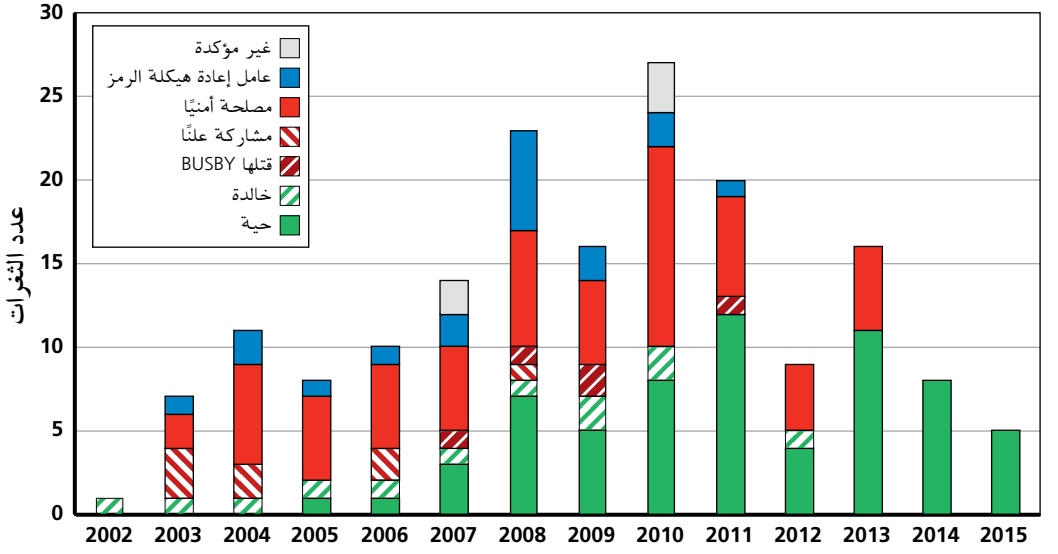
الحالة	الوصف
حية	ثغرة مجهولة للعامة خاصة بالنسخ الراهنة من المنتج، غير مكتشفة ومدونة علنًا من قبل أي شخص آخر (ضمن حدود المعرفة بها)، ومن المرجح أن يكون المضطلعون بأدوار دفاعية يبحثون عنها بنشاط.
خالدة	ثغرة مجهولة للعامة لنسخة المنتج التي أنشأت لأجلها، ولا يخضع المنتج للصيانة (لذا لن يتمّ إصدار إصلاح أمني أبدًا)
مصلحة أمنيًا	ثغرة وجدها طرف ثالث وتعتبر ثغرة أمنية، حيث تم إصدار نشرة و/أو إصلاح و/أو نظام الثغرات الشائعة والانكشاف.
قتلها BUSBY	ثغرة كشفتها جهة خاصة كانت وجدتها عندما أدركت أنه سيتم اكتشاف الثغرة، أو عندما أرادت استخدام ثغرة معينة كأداة تعليمية أو لأغراض تسويقية.
مشاركة علنًا	ثغرة اكتشفها طرف ثالث وجرت مناقشتها علنًا، لكن لم يُعترف بها علنًا على أنها ثغرة أمنية، حيث لم يتم إصدار نشرة و/أو إصلاح و/أو نظام الثغرات الشائعة والانكشاف
عامل إعادة هيكلة الرمز	ثغرة قد تكون مجهولة للعامة للنسخ السابقة من منتج لم تعد قابلة للاستغلال في النسخ الحالية بسبب مراجعات الرمز، وتستمر صيانة المنتج (لذا من الممكن إيجاد إصلاح أمني في المستقبل للنسخ السابقة)

الشكل 3.3
الثغرات التي اكتشفها الباحثون عن الثغرات في BUSBY في كل عام (n = 207)



RAND *RR1751-3.3*

الشكل 3.4
الثغرات التي اكتشفها الباحثون عن الثغرات في BUSBY بحسب الحالة الحالية (n = 207)



RAND *RR1751-3.4*

لم تُكتشف كافة الثغرات المستخدمة لبرمجيات إكسبلويت من قبل باحثي ثغرات يرتبطون بمجموعة بياناتنا. فأحيانًا، تم شراء ثغرة من باحث من طرف ثالث للمساعدة على إكمال ثغرة (يجب ألا ننسى أن برمجية إكسبلويت قد تعتمد على عدة ثغرات لتكون فعالة بالكامل).[6] ضمن مجموعة بياناتنا، جرى شراء تسعة برمجيات إكسبلويت (أي نحو 4%) من باحث من طرف ثالث، حيث كانت اثنتان تنتميان إلى فئة عامل إعادة هيكلة الرمز، وماتت الباقية (تم الإفصاح عن واحدة، في حين تضمنت الأخرى إصلاحات أمنية). ومن بين برمجيات إكسبلويت التي ماتت، كان متوسط عمر البرمجية المستحدثة منها أقل من عام ونصف (521 يومًا).

وقد شهدت بعض السنوات موت عدد أكبر من الثغرات من سنوات أخرى. فقد كان أداء الثغرات التي وجدها الباحثون عن الثغرات في 2010 أسوأ من تلك التي اكتشفت في أي عام آخر. ويمكن شرح ذلك من خلال عمليات تخفيف جديدة بدءًا من 2011 واجهت ثغرات الأعوام السابقة (مثلًا ملفات تعريف الارتباط المكونة)، ما دفع الباحثين عن الثغرات إلى تغيير تقنيات الاستغلال.[7]

[6] على سبيل المثال، قد يحتاج مطور برمجية إكسبلويت إلى ثغرة تسرب المعلومات في جزء من النواة من أجل تخطي عملية التخفيف على غرار تخطيط مساحة عنوان التوزيع العشوائي لمتابعة سائر برمجيات إكسبلويت.

[7] انظر الملحق C "كيف أثرت عمليات التخفيف على قابلية الاستغلال: دراسة حالة الاستغلال في بنية البيانات المكدسة مقابل المكونة" لتحليل بعض البيانات حول التغيرات في أساليب الاستغلال (مثلًا بين ثغرات إفساد الذاكرة القائمة على التكوين ثغرات إفساد الذاكرة القائمة على التكديس).

كيف تموت الثغرات؟

في حين أننا لا نملك بيانات حول كيفية حصول الموت، في ما يتعلق بالثغرات التي نكتشفها أو ثغرات أخرى تستخدم لاستحداث برمجيات إكسبلويت فعالة بالكامل، إلا أننا نجري تقييمات نوعية من عدة باحثين وأفراد تحدثنا إليهم. وقد أجمعوا على أن معظم برمجيات إكسبلويت التابعة لهم (أي ثغراتهم بما أنها باتت معلومة) ماتت بسبب مزيج الرموز، ولسبب آخر أقل شيوعًا ألا وهو إعادة الاكتشاف المستقلة من قبل باحث عن ثغرات يجري تدقيقًا في الرموز. ولا يعتقد أي منهم أن ثغراتهم أو برمجيات إكسبلويت التابعة لهم ماتت أو جرى اكتشافها بسبب استعمال العميل لها في بعض الحملات التشغيلية، أو من خلال تسريب المعلومات (عبر إقدام ويكيليكس Wikileaks أو شادو بروكرز Shadow Brokers على نشر معلومات). وعليه، لا يفصح الباحثون عن الثغرات أيضًا عما يفعله العملاء ببرمجيات إكسبلويت التابعة لهم، أي موعد استخدامها أو كيفية استخدماها أو إن كانوا يستخدمونها أساسًا.

ولأسباب تتعلق بالحساسة والتشغيل وغير ذلك، لا يمكن لعميل غالبًا الإفصاح تحديدًا عن المنتج أو النظام الذي يريد استحداث برمجية إكسبلويت له. انطلاقًا من هذا، يلجأ الباحثون عن الثغرات إلى تخمينات مستنيرة عن حاجات العميل. وفي حين أنه من المثالي بيع كافة برمجيات إكسبلويت المطورة إلى عملاء، تموت بعض برمجيات إكسبلويت في بعض الأحيان دون أن يتم استخدامها. في هذا السياق، أبلغنا فريق بحثي في مجال الثغرات أن 99% من برمجيات إكسبلويت التابعة له تباع إلى عملاء، بينما سجل فريق آخر نسبة 50%، في حين صرح فرد ثالث يتمتع بخبرة في مجال بيع برمجيات إكسبلويت أنه باع جميع برمجيات إكسبلويت التي قام بتطويرها.

2. الديمومة: إلى متى ستبقى الثغرة غير مكتشفة وغير معلنة للعامة؟

قد يساعد تحديد ديمومة برمجية إكسبلويت المحتملة قبل اكتشاف ثغراتها، فضلًا عن برمجيات إكسبلويت التي تعيش لأقصر ولأطول فترة، مطوري برمجيات إكسبلويت على تعديل نماذج أعمالهم وفاحصي الانتشار الدفاعي أو المشغلين الهجوميين على التخطيط لالتزاماتهم.

لقد حاولنا فهم العمر المتوقع لبرمجية إكسبلويت. **ووجدنا أن العمر المتوقع لبرمجيات إكسبلويت هو 6.9 سنوات في المتوسط وأن فترة الصمود الوسطية هي 5.07 سنوات. ونعرّف فترة الحياة القصيرة على أنها الموت (أو عندما تصبح الثغرات مكتشفة ومعلومة) في غضون 1.51 سنة، ما يعادل المئين الخامس والعشرين من توزيع الصمود، في حين تبلغ الحياة الطويلة 9.53 سنوات على الأقل، ما يعادل المئين الخامس والسبعين.**

وفي هذا الجزء من تحليلنا، خفضنا مجموعة بياناتنا من العدد الكامل (n = 207) إلى برمجيات إكسبلويت الحية أو الميتة (إصلاح أمني أو إفصاح علني) وتلك التي لها تاريخ ميلاد (n = 127).[8]

يوفر الجدول F.1 في الملحق F، "أشكال وجداول إضافية" تجزئة لمختلف خصائص برمجيات إكسبلويت.

تحليل الصمود

لقد حاولنا وصف توزيع طول الفترة الممتدة بين ميلاد ووفاة برمجيات إكسبلويت الـ127 الواردة أعلاه بغية تقييم مفاعيل عدة خصائص تتعلق بخطر الموت على مستوى برمجية إكسبلويت، وحساب أمد الحياة المتوقع لبرمجيات إكسبلويت عمومًا ومن حيث خصائصها. ويتطلب ذلك أساليب تحليلية تأخذ في الحسبان الفوارق الزمنية التي تمكنا خلالها من مراقبتها في كل برمجية إكسبلويت. فقد ولدت بعض برمجيات إكسبلويت الموجودة في بياناتنا عام 2003، ما يتيح لنا مراقبتها لأكثر من 10 سنوات قد نتمكن خلالها من تسجيل موتها. في حين ولدت برمجيات أخرى قبل عام أو عامين من إجرائنا التحليل فحسب. وبالتالي، لا يمكننا مراقبة أمد الحياة الكامل لكل برمجية إكسبلويت، أما تلك التي لم ننجح في رصدها عند انتهاء فترة الدراسة، فقد أشرنا إليها على أنها محظورة. هذا وتعتبر البيانات المحظورة شائعة في الدراسات الخاصة بالصحة حيث قد لا يتاح للمحققين مراقبة كل مريض إلى حين حصول حدث مهم. وفي دراستنا، يجري الحظر إداريًا بسبب انتهاء مرحلة جمع البيانات اللازمة للدراسة. وعمومًا، قد يفرض الحظر عند العجز عن متابعة مواضيع ضمن دراسة الوقت إلى الحدث، وحتى في منتصف الطريق في دراسة ما، وربما عند اتخاذ القرار بالابتعاد أو الانسحاب. وهناك مجموعة من الأساليب الإحصائية لتحليل بيانات الوقت إلى الحدث المحظورة، ما يشار إليه جماعيًا بتحليل الصمود.

وصف الصمود مع تقديرات كابلين – ماير

يستخدم تحليل كابلين –ماير (Kaplan-Meier) لتقدير منحنى صمود عدد سكان انطلاقًا من عينة وأرجحية الصمود بناء على حدث مهم (مثلًا إصابة أحد ما بنوبة قلبية) كوظيفة زمنية. في حالتنا، الحدث المهم هو موت برمجية إكسبلويت نتيجة الكشف عن الثغرة التي تتضمنها والإفصاح عنها. يجمع هذا التحليل البيانات المحظورة التي لم تشهد وقوع الحدث المهم (الموت أو الاكتشاف) في كافة المراقبات النموذجية (برمجيات إكسبلويت) قبل انتهاء فترة الدراسة.[9]

وبغية وصف تجربة صمود برمجيات إكسبلويت في إطار العينة التي اخترناها، قمنا ببناء منحنى صمود كابلين –ماير بشكل عام وضمن مجموعات تحددها خصائص برمجيات

[8] لقد ماتت سبع برمجيات إكسبلويت من دون تسجل تاريخ ميلاد لها، إذ اشترى طرف ثالث ثغراتها. لكننا نملك تاريخ نضوج هذه الثغرات (أي تاريخ استحداث برمجية إكسبلويت فعالة بالكامل)، لكن لضمان التطابق مع بياناتنا، قررنا استثناء هذه الثغرات. كما قررنا دمج برمجيات إكسبلويت الخالدة، لكن بما أن تحليل الصمود وأمد الحياة المتوقع يجري عادة على أشخاص أو أشياء من المتوقع أن تموت أو تفشل في مرحلة ما، قمنا بإلغاء برمجيات إكسبلويت الخالدة.

[9] يفترض تحليل كابلين –ماير، وعمومًا تحليل الصمود، أن الحظر لا يحمل معلومات، أي أن برمجيات إكسبلويت المحظورة (لا تزال حية عند انتهاء الدراسة) اختبرت تجربة صمود مماثلة مع مرور الوقت كتلك التي نراقبها إلى حين الموت. للمزيد من المعلومات حول كابلين –ماير، انظر بلاند (Bland) (1998)؛ لام (Lam) (غير مؤرخ)؛ وغويل (Goel) وآخرون (2010).

إكسبلويت. ويخطط منحنى الصمود لاحتمال بقاء ثغرة يومٍ صفر حية على المحور العمودي كوظيفة زمنية منذ الميلاد، تظهر على المحور الأفقي. انطلاقًا من هذه المخططات، حصلنا على وقت الصمود الوسطي، أي الوقت الذي كانت فيه نصف مكونات العينة ماتت كما استخدمنا مخطط كابلين –ماير لتحديد نقاط التجزيء الخاصة بتوزيع فترة الصمود من أجل تحديد العتبات لما قد يعتبر أمد حياة "قصير" أو "طويل". وفي حالتنا، اخترنا المئين الخامس والعشرين والخامس والسبعين، أو الأوقات التي كانت فيها 25% و75% من برمجيات إكسبلويت قد ماتت، باعتبارهما عتبتي أمد الحياة الطويل والقصير.

لقد قمنا بتقدير أمد الحياة المتوقع لبرمجيات إكسبلويت في العينة التي اخترناها، سواء بشكل عام أو وفق خصائص برمجيات إكسبلويت، من خلال حساب المساحة الكامنة في أسفل نطاق الصمود. غير أنه كما يتضح من مخططات كابلين –ماير (الشكلان 3.5 و3.6)، فإن منحنيات الصمود التي قدرناها "لم تنتهِ" بعد. وبناء على ذلك، لم يكن لدينا متسع من الوقت لمتابعة كل برمجية إكسبلويت ومراقبة موتها، ولهذا السبب لا تنزل منحنيات الصمود بالكامل على المحور-x، ما يعني أن احتمال الصمود هو صفر (أي، تنتهي الخطوط أفقيًا وليس عموديًا). ولهذا السبب أيضًا، فالمساحة في أسفل هذه المنحنيات غير محددة. وللسماح بتقدير أمد الحياة المتوقع، استخدمنا نوعًا مختلفًا من نماذج انحسار الصمود، ما يضع افتراضًا توزيعيًا على البيانات التي قد تسمح لنا بتقدير الجزء المتبقي من الوظيفة إلى حين اقترابها من مستوى الصفر بما يكفي للحصول على تقدير مناسب للمساحة الكامنة أسفلها. هذا وقمنا بالتدقيق في عدة أنواع نماذج صمود بارامترية شائعة قد نستخدمها، واخترنا النموذج الذي تناسب مع خصائص بياناتنا على أكمل وجه.

أخيرًا، لتقييم أثر الخصائص الفردية على خطر موت برمجية إكسبلويت، قدرنا استعمال نماذج كوكس للمخاطر النسبية الخاصة انحسار الصمود. وتضع هذه الاستراتيجية للنمذجة فرضيات أقل على البيانات من تلك الموجودة في النمذجة البارامترية المستخدمة لتحديد أمد الحياة، كما تسمح بمرونة أكبر في التعديلات المغايرة من مقارنة منحنيات كابلين –ماير.

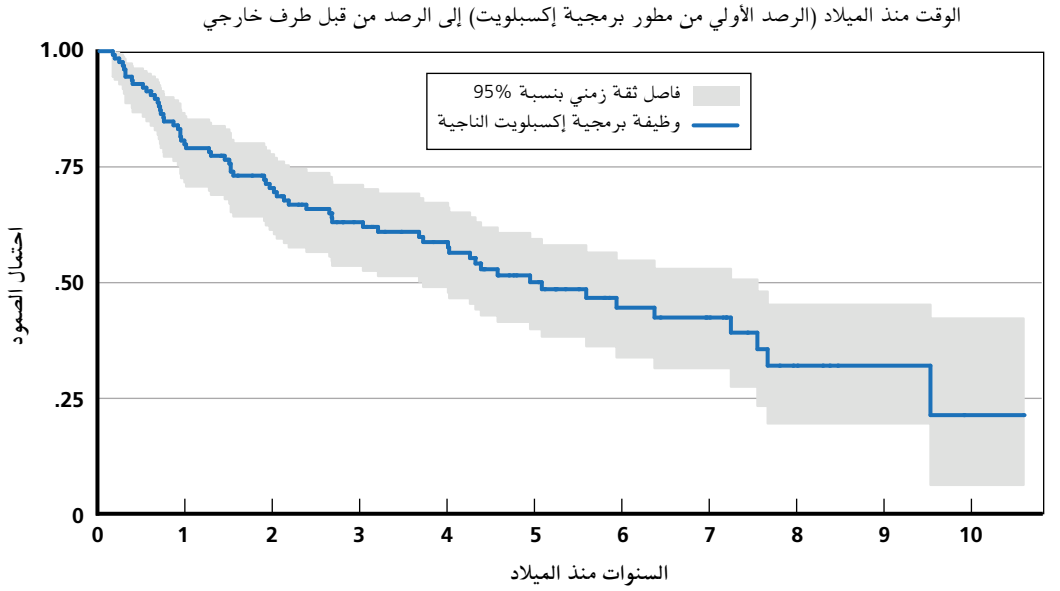
ويظهر الشكل 3.5 تقديرات احتمال الصمود الصادرة عن كابلين –ماير، المحسوبة باستخدام جميع البيانات وبدون أن تتشكل وفق الخصائص. ويظهر الشريط الرمادي الذي يحيط بالخطوط الزرقاء شريط الثقة عند 95%.[10] وتكون فترة الثقة الفاصلة أصغر بالنسبة إلى برمجيات إكسبلويت التي ماتت في أولى مراحلها (بعد سنوات قليلة من الميلاد)، لأنه كان هناك المزيد من نقاط البيانات التي يمكن الاستناد إليها، مع توفير دقة أكبر من حيث تقدير احتمال الصمود.[11]

بعد الرصد الأولي، تتمتع كل برمجية إكسبلويت (وبالتالي ثغراتها) بوقت صمود وسطي يبلغ 5.07 سنوات (فترة الثقة بنسبة 95%: 3.71،7.55). هذا يعني أن احتمال صمود أي برمجية إكسبلويت ضمن مجموعة بياناتنا لمدة خمس سنوات قبل الكشف عن ثغراتها والإفصاح

[10] يمكن تفسير فترات الثقة كما يلي: لأي عمر (منذ الميلاد)، نثق بنسبة 95% بأن احتمال الصمود الفعلي هو ضمن المنطقة الرمادية.

[11] عاشت جميع الثغرات على الأقل لفترة قصيرة، لذا تؤدي الدقة الكبيرة نسبيًا إلى تراجع فترات الثقة. ولا بد من مقارنة هذا مع فترات الثقة الأطول مع ارتفاع عدد السنوات منذ الميلاد، وقد عاش عدد قليل من بسل الاستغلال لهذه الفترة، ما ساهم في توسيع فترة الثقة.

الشكل 3.5
تقديرات احتمال الصمود وفق كابلين –ماير (n = 127)



RAND *RR1751-3.5*

عنها كان 50%. كما أن احتمال الصمود خلال السنوات الأولى لبرمجية إكسبلويت شهد انحدارًا حادًا نسبيًا.[12]

وقد حاولنا أيضًا تحديد عتبات منطقية لوضع خصائص برمجيات إكسبلويت باعتبارها "شابة" أو "قديمة". وحصلنا على المئين الخامس والعشرين والخامس والسبعين من توزيع وقت الصمود أو الأوقات التي كانت فيها 25% و75% من برمجيات إكسبلويت ماتت على التوالي.[13] وباستخدام نقاط التجزيء هذه، نعتبر أن أمد حياة برمجيات إكسبلويت قصير إن ماتت عند 1.51 سنة أو قبل، ما يعادل الوقت الذي كانت فيه 25% منها ماتت. أما برمجيات إكسبلويت التي تدوم 9.53 سنوات على الأقل، فتكون عاشت لفترة موازية أو أطول من نسبة 75% من العينة، لذا يعتبر أمد حياتها طويلًا.

ولتقييم إن كان الصمود يختلف بحسب خصائص برمجية إكسبلويت، قمنا بتخطيط منحنيات كابلين –ماير بشكل منفصل لخصائص برمجية إكسبلويت الأربعة التالية: نوع الثغرة، والمنصة المتضررة، ونوع رمز المصدر، ونوع فئة برمجية إكسبلويت. ويوفر الملحق

[12] يظهر ذلك من خلال جدول الحياة (الجدول ح.2) ووظيفة الخطر الممهدة التي تناسب مخطط جدول الحياة (الشكل ح.2).

[13] نقر أنه كان بإمكاننا بسهولة اختيار نسبة 10% كذروة وقاع، أو أي نسبة أخرى.

H، "المسرد" تعريفات ومعلومات إضافية حول مختلف أنواع الثغرات وأنواع فئات برمجيات إكسبلويت:[14]

1. **نوع الثغرة:** فساد الذاكرة، سوء إدارة الذاكرة، المنطق، مختلطة، غير ذلك.
2. **المنصة المتضررة:** لينكس (Linux)، ماك أو.إس (OSX)، مفتوحة المصدر، بي إتش بي (PHP)، قائمة على يونكس (Unix-based) (باستثناء لينكس)، ويندوز (Windows)، مختلطة أو غير ذلك.
3. **نوع رمز المصدر:** مفتوح، مغلق، مختلط.
4. **نوع فئة برمجية إكسبلويت:** من جانب العميل، محلية، عن بعد، مختلطة أو غير ذلك.

لقد أجرينا اختبار تصنيف لوغارتمي لكل خاصية (نوع الثغرة والمنصة المتضررة ونوع رمز المصدر ونوع فئة برمجية إكسبلويت)، لاختبار الفوارق في منحنيات كابلين–ماير في مختلف المستويات (مثلًا خاصية نوع رمز المصدر المفتوح أو المغلق أو المختلط). يظهر الشكل 3.6 منحنيات كابلين–ماير لكل خاصية وقيمة p في اختبار التصنيف اللوغارتمي انطلاقًا من مقارنة المنحنيات ضمن كل لوحة لمعرفة ما إذا كانت المنحنيات مختلفة من الناحية الإحصائية.[15]

لم نلحظ إحصائيًا فوارق كبيرة في الصمود في أي من خصائصنا الأربع. ويكون رصد الفوارق في المنحنيات أصعب عندما تتمتع الخاصية بعدة مستويات، لأن عدد الإخفاقات يتراجع أكثر فأكثر مع تقسيم البيانات إلى المزيد من المستويات.

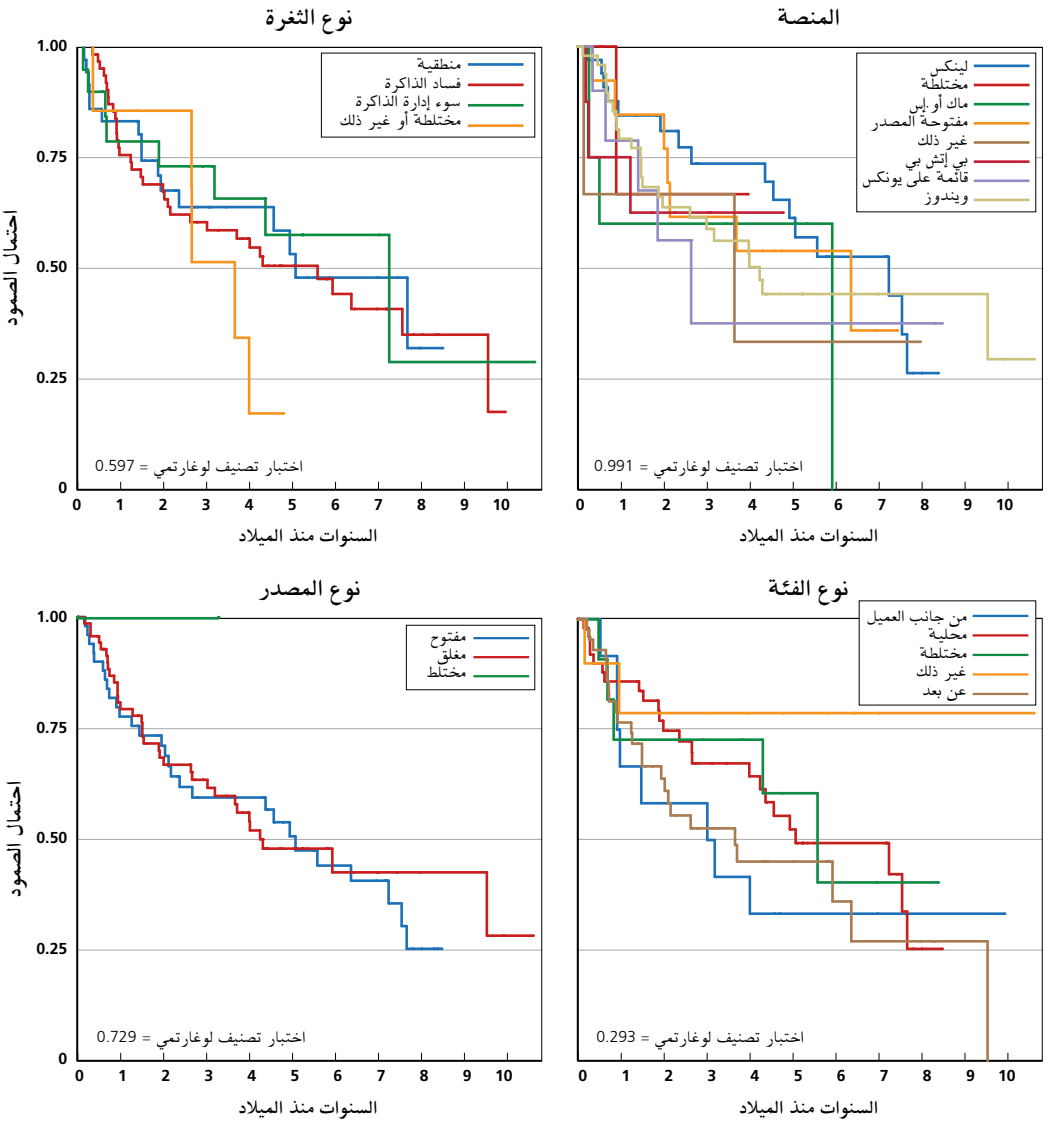
لا تظهر المنحنيات في لوحة **نوع الثغرة** مؤشرات على فوارق مهمة ضمن مستوياتها (p = 0.597). أما بالنسبة إلى **المنصة المتضررة**، فيصعب العدد الكبير للفئات رصد الفوارق في مختلف المستويات. وقد قمنا بالتدقيق عبر مقارنة برمجيات إكسبلويت قائمة على منصة لينكس مع المنصات الأخرى مجتمعة، حيث يشير المخطط إلى أن برمجيات إكسبلويت القائمة على منصة لينكس قد تميل إلى الصمود لفترة أطول، رغم أن قيمة الفرق كانت لا تذكر عند 0.357. في مخطط **نوع المصدر**، تعتبر منحنيات برمجيات إكسبلويت المفتوحة والمغلقة المصدر متطابقة تقريبًا، ومتماشية مع استنتاجات محققين آخرين (p = 0.729) (ألتكينكيمر Altkinkemer وريس Reese وسريدار Sridhar، 2008؛ شراين Schryen وريتش Rich، 2010). تجدر الملاحظة أن منحنى الفئة "المختلطة" في هذه اللوحة لا يتخذ مسارًا تنازليًا أبدًا، كما أننا لم نلحظ حالات موت في أوساط برمجيات إكسبلويت "المختلطة". وأخيرًا، ضمن لوحة **نوع الفئة**، نواجه مجددًا تحدي المقارنة ضمن عدد كبير من الفئات ووجدنا أنه لا فرق يُذكر في الصمود في مختلف أنواع الفئات (p = 0.293). غير أن هذا الرقم يتضمن اقتراحًا

[14] ظهرت بعض الخصائص على أنها "مختلطة" أو "غير ذلك". وتشير الخصائص "المختلطة" إلى برمجيات إكسبلويت التي تقوم على عدة ثغرات مرتبطة ببعضها البعض (لنوع الثغرة) أو عدة منصات أو عدة أنواع من رموز المصدر أو عدة فئات برمجيات إكسبلويت. أما الخصائص "الأخرى" فتشير إلى فئات تضمنت عددًا قليلًا جدًا من نقاط البيانات ليتم أخذها في الحسبان بحدّ ذاتها في هذا التحليل، لذا تمّ جمعها.

[15] تشير المنحنيات التي تنتهي بخط أفقي إلى أن القيمة الأخيرة المسجلة كانت حية، في حين تشير المنحنيات التي تنتهي بخط عمودي إلى أن القيمة الأخيرة المسجلة كانت ميتة (أي تم الإفصاح عنها للعامة) وأنه ما من دليل على الصمود المتواصل لهذا العنصر على وجه الخصوص.

الشكل 3.6

منحنيات كابلين–ماير لنوع الثغرات والمنصة المتضررة ونوع رمز المصدر ونوع فئة برمجية إكسبلويت (n = 127)



RAND *RR1751-3.6*

يتمثل بأن تقسيم هذا المتغير إلى فئتين – أي جمع برمجيات إكسبلويت من جانب العميل وتلك عن بعد ومقارنتها مع برمجيات إكسبلويت المجمعة المحلية والمختلطة والأخرى – قد يولد مجموعات مختلفة. وعلى سبيل الاستكشاف، شكلنا هذه المجموعات ووجدنا أنها لا تختلف بشكل كبير ($p = 0.068$، المخطط غير ظاهر)، رغم أن قيمة p منخفضة بما يكفي لتشير إلى دراسة هذه المسألة المحتملة بشكل مفصل.

وقد يعزى غياب مراقبة الفوارق المهمة من الناحية الإحصائية من حيث الصمود في الخصائص الأربع إلى عدم وجود أي اختلاف في الواقع، رغم أنه قد يعزى أيضًا إلى غياب البيانات. وعليه، **يوفر تحليلنا توجيهات حول الفرضيات التي يجب اختبارها في التحليلات المستقبلية، لا سيما لمراقبة ديمومة الثغرات الخاصة بلينكس مقارنة مع منصات أخرى، ولتأكيد تشابه ديمومة الثغرات لنوعي رمز المصدر المفتوح والمغلق، وللتحقيق في أهمية جمع برمجيات إكسبلويت من جانب العميل وبرمجيات إكسبلويت عن بعد مقارنة مع جمع برمجيات إكسبلويت المحلية والمختلطة وغير ذلك.**

أمد الحياة المتوقع

إلى جانب البحث في أوقات الصمود المتوسطة والطويلة والقصيرة، أردنا مراقبة تجربة صمود بياناتنا بطريقة مختلفة وهي: أمد الحياة المتوقع. يمكن تقدير أمد الحياة المتوقع مباشرة من خلال حساب المساحة الكامنة أسفلها. إنها عملية سهلة نسبيًا، شرط أن نراقب الفشل في كل موضوع (أي، اكتشاف/موت كافة برمجية إكسبلويت لاستغلال ثغرات يوم الصفر) لخفض المنحنى إلى الصفر. غير أن منحنيات كابلين–ماير الخاصة بنا لا تبلغ الصفر بما أننا لم نشهد موت برمجيات إكسبلويت الـ27 بكاملها، ما يجعل من المستحيل حساب المساحة الكامنة أسفل منحنى كابلين– ماير مباشرةً.

وفي حالات مماثلة، يمكن استخدام أدوات أخرى لتحليل الصمود تسمح بتقدير منحنى صمود كامل، حتى عندما يتم فرض حظر على موعد المراقبة الأخير، شرط أن نكون راغبين في وضع فرضيات بارامترية على توزيع فترة الصمود. كما تفحصنا النموذج الأسي (exponential) ونموذج وايبول (Weibull) ونموذج الصمود اللوغارتمي الطبيعي (log-normal survival) بما أنها تلائم بياناتنا من خلال مقارنة مخططاتها المقدرة للصمود مع مخططات كابلين–ماير التي بنيناها مسبقًا.[16] تناسب كل من هذه النماذج بياناتنا بشكل منطقي أيضًا، وتوفر منحنيات تقع ضمن شريط الثقة في منحنى كابلين–ماير الإجمالي في الشكل 3.5. لكننا شعرنا أن النموذج الأسي كان الخيار الأفضل لبياناتنا، استنادًا إلى فرضيته المتمثلة بوظيفة الخطورة الدائمة، وهي وظيفة تعكسها بياناتنا على ما يبدو. تجدون المزيد من التفاصيل حول وظيفة الخطورة والمخططات في تقديراتنا غير المعدلة للخطورة ومقارنات النماذج البرامترية في الملحق F، "أشكال وجداول إضافية".

ومن خلال استخدام نموذجنا للانحسار الأسي، تمكنا من الحصول على تقديرات لمنحنى الصمود القائم على نموذج تمتد اعتباطيًا قرب المحور x، ما سمح لنا بحساب المساحة الكامنة أسفلها، وهي تقديراتنا غير المعدلة لأمد حياة برمجية إكسبلويت متوسطة ضمن العينة التي

[16] راجع الشكل F.1 في الملحق F، "أشكال وجداول إضافية".

الجدول 3.2
تقديرات أمد الحياة العامة ومن حيث خصائص برمجيات إكسبلويت المستقاة من نماذج الصمود الأسي غير المعدلة (n = 127)

خاصية برمجية إكسبلويت	تقديرات أمد الحياة وفق النموذج الأسي (الفاصل الزمني للثقة 95%)
عامة	6.90 (5.39, 8.84)
نوع الثغرة	
منطقية	7.20 (4.41, 11.75)
فساد الذاكرة	6.75 (4.83, 9.45)
سوء إدارة الذاكرة	8.81 (4.41, 17.55)
مختلطة أو غير ذلك	3.92 (1.63, 9.40)
المنصة المتضررة	
لينكس	8.62 (5.28, 14.06)
مختلطة	6.90 (0.98, 42.63)
ماك أو.إس	5.36 (2.02, 14.27)
مفتوحة المصدر	7.32 (3.49, 15.32)
غير ذلك	5.90 (1.48, 23.25)
بي أتش بي	5.59 (1.81, 17.27)
قائمة على يونكس	5.95 (2.48, 14.27)
ويندوز	6.36 (4.30, 9.42)
نوع المصدر	
مفتوح	6.51 (4.50, 9.43)
مقفل	6.93 (4.95, 9.70)
مختلط	–
نوع فئة برمجية إكسبلويت	
من جانب العميل	5.16 (2.58, 10.32)
محلية	7.63 (5.12, 11.39)
مختلطة	8.71 (3.63, 20.75)
غير ذلك	21.48 (5.43, 59.36)
عن بعد	5.14 (3.45, 7.67)

اخترناها. وقمنا بملاءمة أربعة نماذج أسية إضافية، مع تعديلها لتتواءم مع الخصائص الأربع لبرمجية إكسبلويت (نوع الثغرة والمنصة المتضررة ونوع المصدر ونوع فئة الاستغلال) على التوالي من أجل الحصول على أمد حياة برمجيات إكسبلويت ضمن كل مستوى من الخصائص. يذكر أن الجدول 3.2 يمثل تقديرات أمد الحياة المتوقع، سواء العام ومن حيث الخصائص المحددة.

ويكمّل أمد الحياة نتائج منحنيات كابلين–ماير بطريقة حدسية. فكلما كانت وظيفة الصامد أكثر انحدارًا، كان أمد الحياة المتوقع أقصر. كما أنه كلما راقبنا المزيد من الأحداث باعتبارها "خطوات" تنازلية على مخطط كابلين–ماير، كلما ضاق فاصل الثقة الزمني المحيط بتقديرات أمد الحياة المتوقع. وتختلف دقة تقديراتنا لأمد الحياة المتوقع بشكل كبير بحسب الخصائص، فهناك فئات لاحظنا فيها أن العديد من الأحداث وفرت فواصل ثقة زمنية أضيق حول تقديراتنا المحددة (مثلًا، ثغرات وبرمجيات إكسبلويت عن بعد والمحلية في ويندوز)، وفئات لم نلحظ الكثير من الأحداث فيها (مثلًا انظر إلى أنواع برمجيات إكسبلويت "المختلطة" و"غير ذلك") إنما تضمنت فترات ثقة أكبر بكثير.[17] وبافتراض أن ما نملك هو عينة نموذجية لبرمجيات إكسبلويت، وجدنا أن برمجية إكسبلويت المتوسطة (وبالتالي ثغراتها) يتوقع أن تعيش لفترة 6.90 سنوات (مع فاصل ثقة زمني بنسبة 95%: 5.39 إلى 8.84 سنوات).[18]

نموذج انحسار الخطورة النسبية

من أجل تقييم ما إذا كانت أي خاصية محددة لبرمجية إكسبلويت تساهم في خفض أو رفع احتمال الموت، ما يزيد أو يقلل أمد الحياة، قمنا باعتماد نماذج كوكس (Cox) للصمود الخاصة بالخطورة النسبية.

لقد استخدمنا نموذج كوكس لانحسار الخطورة النسبية من أجل تقدير معدل الخطورة المرافق لكل مستوى في كل خاصية بالمقارنة مع فئتها المرجعية. واعتمدنا نماذج غير معدلة لكل من الخصائص الأربع، فضلًا عن تعديل نموذج معدل بالكامل للخصائص الأربع بالتزامن، ونموذج معدل جزئيًا استثنى المنصة المتضررة. يوفر الجدول F.3 والنص المرافق في الملحق F، "أشكال وجداول إضافية" المزيد من التفاصيل عن المنهجيات المستخدمة وقيم كل نموذج.

وبالتماشي مع نتائج كابلين–ماير، لا يبدو أن هناك أي خصائص قد تشكل "دليلًا" على حياة طويلة أو قصيرة، رغم أن استنتاجاتنا مفيدة لوضع فرضيات. فقد كانت خاصية "الدليل" تشير إلى أن ثغرة أو برمجية إكسبلويت التي تستغلها قد تدوم لفترة طويلة أو قصيرة، وربما تساعد تلك التي تتعامل مع عملية معالجة الثغرات على تحسين ما يجب الاحتفاظ به وما يجب كشفه علنًا. مثلًا، إذا أظهرت الإحصاءات أن جميع برمجيات إكسبلويت القائمة على ثغرات فساد الذاكرة تتمتع باحتمالات صمود أدنى مع مرور الوقت، فربما يجب أن تكون هذه البرمجيات من نوع الثغرات التي يتم كشفها علنًا وإصلاحها (أي لا يتم تخزينها)، لأن الذين يرغبون في استغلالها لإتمام عملياتهم الخاصة قد يملكون وقتًا محدودًا فقط، وبالتالي قد يصبح

[17] تظهر فواصل الثقة الزمنية نوعًا ما دقة تقديراتنا لأمد حياة كل فرد. وبالتالي، تمكنا بنسبة 95% من التصريح بأن أمد حياة كل خاصية يكمن ضمن فواصل الثقة الزمنية كما يظهر.

[18] يعني فاصل ثقة زمني بنسبة 95% تتراوح بين 5.39 و8.84 سنوات أننا على ثقة بنسبة 95% بأن متوسط أمد حياة برمجية إكسبلويت مشابهة لتلك الموجودة في العينة التي اخترناها يتراوح بين 5.39 و8.84 سنوات.

عدد الأشخاص المعرضين للخطر أكبر نظرًا إلى وتيرة الاكتشاف السريعة.[19] وعلى نحو مماثل، إذا كان نوع أو جانب محدد من الثغرة يشير إلى فترة طويلة، قد يكون التخزين خيارًا مبررًا.

وفي ما يتعلق ببياناتنا، بالنسبة إلى كل خاصية، تكون القيم p من اختبارات والد (Wald) لتقديرات معامل الانحسار مرتفعة جدًا بحيث أنها لا تحدث أي فوارق مهمة عند أي مستوى، مع تعديل للخصائص الأخرى أو بدون ذلك. وفي غياب المزيد من البيانات، لا يمكننا أن نحدد إحصائيًا إن كانت أي من خصائص برمجية إكسبلويت تشير إلى حياة طويلة أو قصيرة. فقيم p المرتفعة قد تكون نتيجة غياب فعلي للأثر، أو نتيجة عدد محدود من حالات الموت المسجلة، بسبب حجم العينة الضئيل. ومن شأن المزيد من البيانات وفترة متابعة أطول توفير حلول مهمة إحصائيًا عند إجراء دراسة إضافية.

3. معدل الاصطدام: ما هو احتمال أن يكتشف الآخرون الثغرة ويفصحون عنها؟

يعتبر فهم ديمومة ثغرة معينة مهمًا، وكذلك إدراك احتمال اكتشاف الثغرة من قبل باحثين آخرين، بغض النظر عن طول الفترة التي كانت خلالها الثغرة حية.

قد يشير معدل اصطدام مرتفع إلى أن الثغرات التي اكتشفها باحثون خاصون وصائدو أخطاء عامون سهلة الاكتشاف نسبيًا، ما يعزز الحجة القائلة بالإفصاح عن أي ثغرة مكتشفة.[20] وعلى نحو مماثل، قد يشير معدل اصطدام منخفض إلى أن الثغرات التي وجدها باحث ما ستبقى عصية على الباحثين الآخرين.

عوضًا عن ذلك، قد تسفر الثغرات التي يصعب اكتشافها وتكون منغمسة في قاعدة الرموز عن تداخل كبير في حال استخدم عدد من الباحثين عن الثغرات تقنيات اصطياد الأخطاء نفسها.[21]

وتقوم منظمة تحتفظ بالثغرات أو برمجيات إكسبلويت على الأرجح بتقييم مخزوناتها خلال فترات منتظمة، حيث تلغي تلك التي جرى اكتشافها والإفصاح عنها علنًا وتحتفظ بتلك التي لا تزال حية أو يمكن استخدامها لهجمات يوم صفر. في حال التخلص من برمجيات إكسبلويت ماتت لتوها خلال فترات منتظمة، عندها تمثل النسبة المئوية لبرمجيات إكسبلويت الميتة في نهاية كل فترة معدل التداخل أو الاصطدام.

[19] حتمًا، يعتمد ذلك على المنصة والنظام المستهدف المتضرر. قد يحدث أنه حتى إن كان يمكن استغلال ثغرة بسرعة وتتمتع بأمد حياة قصير، قد يكون استعمالها لفترة قصيرة مفيدًا.

[20] لا تعني قاعدة رموز أو منتج يتضمن العديد من الثغرات المكتشفة (أي نسبة تداخل مرتفعة وكمية كبيرة) بالضرورة أن قاعدة رموز أو منتج محدد هو المهاجم السيئ أو الأسوأ: فالحصة السوقية قد تدفع بصائدي الأخطاء والباحثين عن الثغرات إلى البحث عن قواعد بيانات معينة. وقد أظهرت بعض الأبحاث (مايلارت Maillart وغيره، 2016) أن صائدي الأخطاء غالبًا ما يتنقلون بين منتج وآخر بحثًا عن المردود الأسرع (وأحيانًا الأرخص).

[21] غالبًا ما يؤدي تداخل الثغرات في قاعدة الرموز إلى مناقشة الندرة مقابل الكثافة إلى جانب نضوب الثغرات. ويشكل هذا الموضوع محور نقاش محتدم تناوله وكتب عنه الكثيرون. ويعتقد شناير (Schneier) (2014) أن الثغرات كثيرة (ما قد يفسره المرء على أنه كثافة)، وعليه فإن التداخل صغير نسبيًا. من جهته قال غير (Geer) (2014) إن الثغرات نادرة بما يكفي ما يدفع بالحكومة الأمريكية إلى احتكار سوق شراء الأخطاء. هذا ولفت أوزمانت (Ozment) وشيشتر (Schechter) (2006) إلى أن كثافة الثغرات في نظام أوبن بي.إس.دي (OpenBSD) التشغيلي تتراوح بين 0 و0.033 ثغرة مسجلة لكل ألف خط رمز.

هذا وتركز الدراسات الخاصة بمعدل الاصطدام عمومًا على الثغرات المستخدمة في برامج مكافآت الثغرات أو تلك المكتشفة علنًا والمسجلة في إطار قاعدة رموز. وقد خلص فينيفتر (Finifter) وأكاوي (Akhawe) وواغنر (Wagner) (2013) إلى أن 2.25-5% من جميع الثغرات المسجلة ضمن برامج مكافآت الثغرات قد اكتشفها آخرون. وتشير تقديرات مؤسسة RAND من حيث المقدار الأسي إلى أن احتمال أن يكتشف طرفان ثغرة خلال سنة واحدة يناهز 10% (ليبيكي Libicki وأبلون Ablon وويب Webb، 2015). وقد استحدث الباحثون في 2015 نموذجًا إرشاديًا وجد تداخلًا بنسبة 9% في برمجية لم يتمّ اختبارها أمنيًا و0.08% في برمجية أكثر أمانًا (موسوريس Moussouris وسيغيل Siegel، 2015).

ضمن بياناتنا تم رصد نحو 40% من برمجيات إكسبلويت والإفصاح عنها علنًا. لقد حسبنا هذه القيمة من خلال مراكمة كافة برمجيات إكسبلويت والاحتفاظ بها طيلة عمر مجموعة بياناتنا البالغ 14 عامًا، بغض النظر عن موت برمجية إكسبلويت أو ثغرة.

لكن على الأرجح قد تقوم منظمة بإزالة الثغرات الميتة من مخزونها خلال فواصل زمنية منتظمة. وباعتماد فواصل زمنية من 365 يومًا و90 يومًا (إلى جانب 30 يومًا ويوم واحد)، أخضعنا بياناتنا لتحليل الحساسية، من أجل تقييم النسبة المئوية للثغرات التي ماتت خلال فواصل زمنية معينة. وفي كل فاصل، أزلنا الثغرات التي ماتت في الفترة السابقة. وقمنا بفصل كل برمجية إكسبلويت إلى أربع فئات رئيسية: (1) حية و(2) عامل إعادة هيكلة الرمز و(3) ميتة و(4) مجهولة. وتمثل برمجيات إكسبلويت التي ماتت في نهاية فترة زمنية محددة اصطدامات. **هذا وتتغير معدلات الاصطدام بشكل كبير وفق الفاصل الزمني المستخدم. ففي حين وجدنا معدل اصطدام بنسبة تناهز 40% خلال مدة بياناتنا البالغة 14 عامًا، أسفر تضييق الفواصل الزمنية عن معدل اصطدام بنسبة 5.76% (قيمة وسطية) لفاصل زمني من 365 يومًا (انحرف معياري بنسبة 6.79%)، ومعدل اصطدام بنسبة 0.87% (قيمة وسيطة) لمدة 90 يومًا (انحراف معياري بنسبة 5.3%). بعبارة أخرى، لمخزون محدد من ثغرات يوم الصفر، اكتشف الآخرون نحو 5.7% منها وأفصحوا عنها في غضون عام واحد.**[22]

تظهر القيم في الشكلين 3.7 و3.F والجدول F.4 في الملحق F، "أشكال وجداول إضافية".

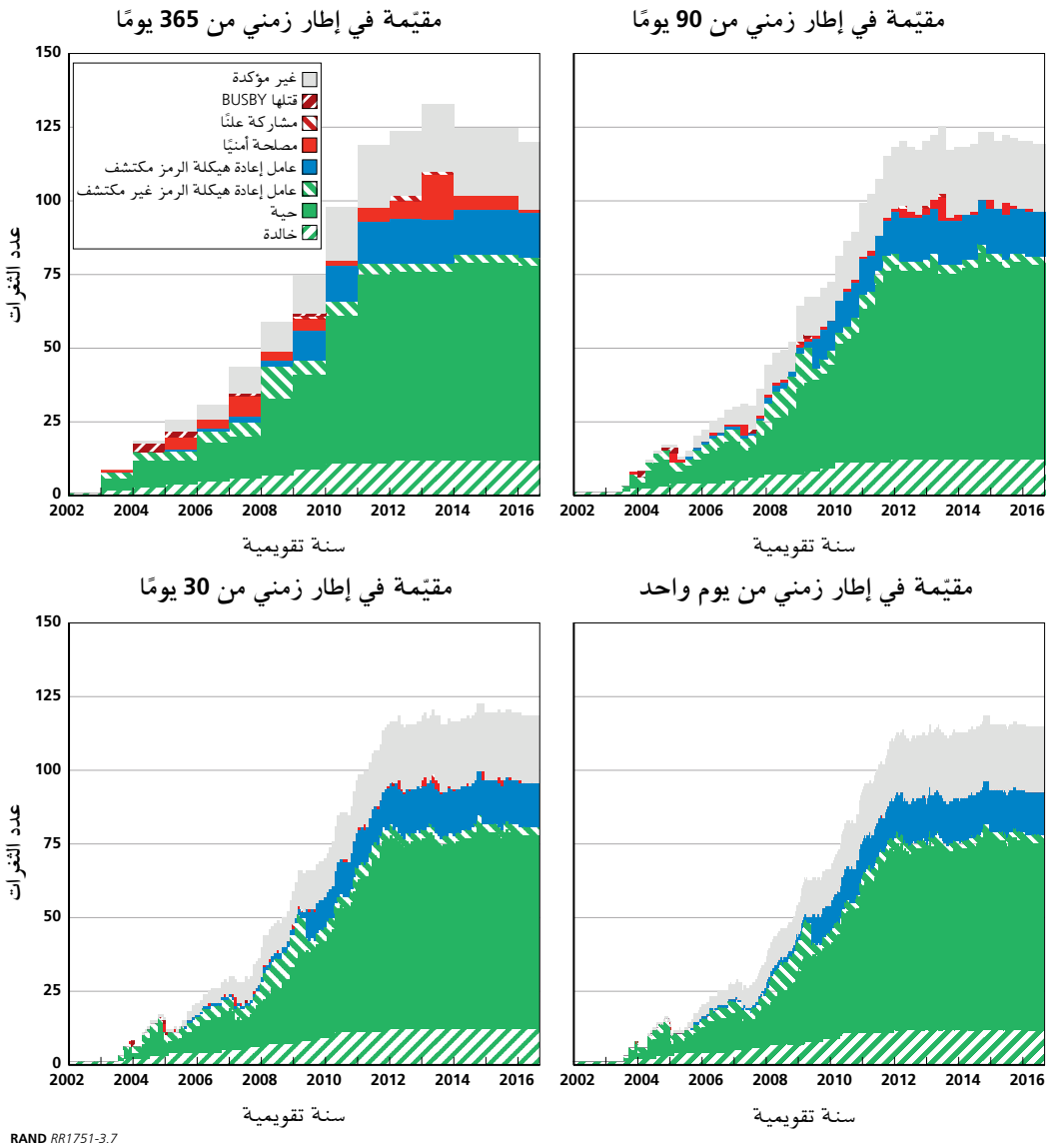
أوقات الاصطدام مع الثغرة نفسها

قد يرغب المرء أيضًا بدراسة الفترة التي كانت خلالها الثغرة معلومة للجهة الخاصة قبل الكشف عنها والإفصاح عنها للرأي العام، أو في ما يخص بياناتنا، الفترة التي مرت قبل اكتشاف الثغرات التي نعرفها والإفصاح عنها علنًا.

وفي حين أنه يصح أن العديد، أو حتى الغالبية كما يقول البعض، من استعمالات برمجيات إكسبلويت تتركز على الثغرات المكتشفة أساسًا والتي لم تعد تصنف كثغرة يوم صفر (بيلج Bilge ودوميتراس Dumitras، 2012) قد يكون من المفيد معرفة احتمال بقاء ثغرة يوم صفر بالفعل ثغرة يوم صفر ولكم من الوقت. على وجه الخصوص، قد يرغب أولئك الذين يستخدمون برمجيات إكسبلويت ثغرات يوم الصفر لأغراض هجومية أو دفاعية، على غرار اختبار الانتشار

[22] في حين قمنا بتحليل الفواصل الزمنية المؤلفة من 365 و90 و30 و1 يوم، وفرنا نتائج للفواصل الزمنية التي تضم 365 و90 يومًا فقط. وقمنا بحساب القيم من خلال النسبة الوسيطة لحالات الوفاة في كل فاصل زمني. ومع تراجع حجم الإطار الزمني، برزت عدة أطر لم تشهد أي حدث، ما جعل القيمة الوسطية عند صفر. فقد كانت القيمة المتوسطة صفر للفاصلين الزمنيين المؤلفين من 30 يومًا ويوم واحد.

الشكل 3.7
أنواع الثغرات مع مرور الوقت (n = 192)



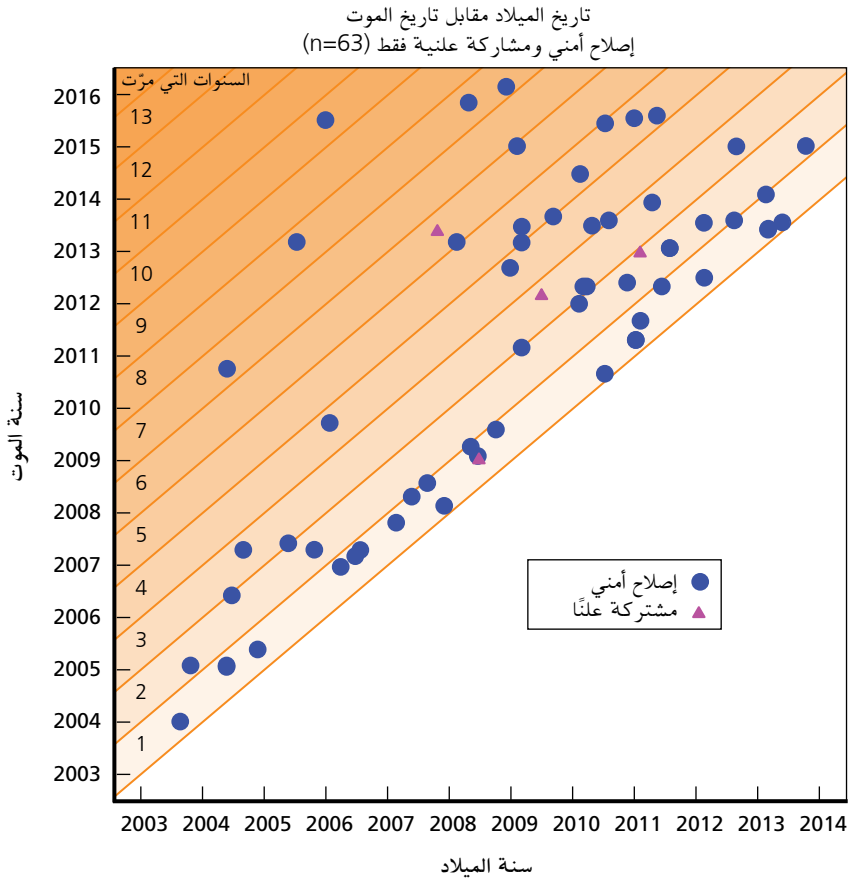
أو تجربة أداة رصد جديدة، في معرفة الفترة التي قد تمر قبل أن يكتشف الجمهور المستهدف أمرهم خلال حملة. كما قد يرغب الباحثون عن الثغرات ومطورو برمجيات إكسبلويت في معرفة الفترة التي تدوم خلالها برمجيات إكسبلويت التابعة لهم عامة لأغراض الأعمال أو التسويق.[23]

[23] على سبيل المثال، يجب استخدام البيانات لتحديد الأسعار أو إبلاغ العملاء بالقدرات المتوقعة.

لقد راقبنا هذه الثغرات التي اكتشفها شخص آخر بهدف التحقيق في احتمال الاصطدام ولمراقبة معدل الاضمحلال. أما بالنسبة إلى بياناتنا، "فالاكتشاف من قبل آخرين" يترجم في حالة "الإصلاح الأمني" أو "المشاركة علنًا"، وهما حالتان لدينا تواريخ الميلاد والوفاة الخاصة بهما.[24]

يظهر الشكل 3.8 الفارق الزمني بين موعد اكتشاف باحثي الثغرات في BUSBY الثغرة المستخدمة في برمجية إكسبلويت التابعة لـBUSBY وموعد اكتشافها وكشفها للعامة من قبل طرف ثالث مستقل.

الشكل 3.8
توقيت التداخل بين موعد اكتشاف جهتين مستقلتين لثغرة (2002 - 2016) (n=63)



RAND *RR1751-3.8*

[24] في حين أن "عامل إعادة هيكلة الرمز" أو "قتلها BUSBY" هما نوعا وفاة أيضًا، إلا أنهما لا يشيران إلى اكتشف الثغرة (في حالة عامل إعادة هيكلة الرمز) أو اكتشافها من قبل شخص آخر (في حالة قتلها BUSBY).

تمثل كل منطقة مائلة مظللة عامًا واحدًا: تتضمن منطقة القاع المائلة المظللة (الرقم 1) نقاطًا هي برمجيات إكسبلويت اكتشفها وأفصح عنها شخص من خارج BUSBY في غضون عام بعد اكتشافها الأولي من قبل BUSBY. وتحمل كل منطقة مائلة فوق المنطقة الأولى من الجانب الأيسر عدد السنوات التي مرت بين الاكتشاف الأولي من قبل BUSBY والاكتشاف والإفصاح المستقل اللاحق.[25] وبالتالي، فإن العدد أقل للنقاط في الأشرطة المائلة العليا هو أكثر انخفاضًا بشكل اصطناعي مما كان المخطط ليظهره، لو كنا نملك الوقت الكافي للمتابعة.

وبما أن هذا التحليل لا يأخذ في الحسبان الحظر، لا يمكننا استخلاص أي استنتاجات حول معدل التداخل. وعليه، **يمكن للمرء أن يقوم بالمراقبة النوعية المتمثلة بأنه، بين عامي 2002 و2016، أظهرت البيانات على ما يبدو أنه بالنسبة إلى برمجيات إكسبلويت التي ماتت (أي التي اكتشفتها أطراف مستقلة)، تحصل حالات الوفاة بسرعة نسبية وغالبًا في غضون عام واحد.** وقد يشير المخطط أيضًا إلى أن معدل الاكتشاف لا يكون متسقًا كل عام. فعلى سبيل المثال يبدو أنه منذ مطلع 2008، عاش عدد من برمجيات إكسبلويت التي اكتشفها الباحثون عن الثغرات في BUSBY لفترة أطول من تلك المكتشفة خلال السنوات السابقة. لكن السبب لا يزال غير واضح. فقد يعزى إلى أن الباحثين طوروا تقنية جديدة أو اكتشفوا مجموعة جديدة من برمجيات إكسبلويت مختلفة عن سابقاتها، أو برزت عمليات التخفيف التي منعت برمجيات إكسبلويت من أن تعيش لفترة طويلة قبل 2008، أو تمكن عضو جديد في الفريق من اكتشاف ثغرات موجودة عند مستوى عميق في الرمز أو المنتجات التي سجلت درجات مرتفعة كثيرًا أساسًا أو إلى سبب آخر مجهول. ولنكون واضحين إنها مجرد تخمينات، فقد نشهد في المستقبل الكشف والإفصاح بشكل مستقل عن ثغرات وجدها BUSBY في 2005، ما قد يثير الشكوك بشأن لائحة الأسباب المحتملة.

4. التكلفة: ما هي تكلفة تطوير برمجية إكسبلويت لاستغلال ثغرة؟

التخطيط لبناء مخزون: فهم تكلفة الحفاظ على برمجيات إكسبلويت جاهزة للاستعمال
إذا رغب شخص أو منظمة في الحفاظ على مخزون جاهز للاستعمال من الثغرات، من المفيد معرفة ما قد يكون ضروريًا لتغذية المخزون بالكامل والحفاظ عليه، ولا سيما تكلفة تطوير برمجية إكسبلويت وما إذا كان اللجوء إلى مصادر داخلية أو خارجية هو الخيار الأفضل.[26]

قد تعتمد تكلفة برمجية إكسبلويت، وعلى نحو متصل، قيمتها أو سعرها على عدة عوامل قد تؤثر على التكلفة قبل تطوير برمجية إكسبلويت أو بالتزامن معه. وتشمل هذه العوامل الوقت الضروري لاكتشاف ثغرة يوم صفر قابلة للنمو (وقت البحث)، والوقت اللازم لتطوير

[25] يظهر الرسم البياني الولادات التي حدثت خلال 2013، لكن هذا لا يعني أن محللي BUSBY توقفوا عن اكتشاف الثغرات واستغلالها في ذلك العام. بل الأحرى أننا لا نملك تواريخ ميلاد لحسابها. كما هناك بعض برمجيات إكسبلويت من BUSBY لعام 2014 و2015 و2016 لا تشملها مجموعة بياناتنا بسبب الحساسات التشغيلية الإضافية. هذا وتضم مجموعة بياناتنا خمس ثغرات مبيعة إلى BUSBY من قبل أطراف خارجية ثالثة في 2014 وما بعده، ماتت جميعها، لذا لم نتمكن من تحديد تواريخ اكتشاف برمجيات إكسبلويت.

[26] حتمًا في كثير من الأحيان لا يتعمد استعمال برمجية إكسبلويت أو فائدتها على كون ثغراتها ثغرات يوم صفر. لكن دعمًا للحجج المقدمة، نقوم بدراسة الحالات التي قد يرغب المرء فيها في وجود ثغرات يوم صفر.

برمجية إكسبلويت للاستفادة من ثغرة يوم صفر (وقت تطوير برمجية إكسبلويت)، وتكلفة شراء جهاز أو رمز للمراجعة أو الاستحواذ عليه، والوقت الضروري لإعداد مختبر تجارب وتكلفة البنية التحتية أو الأدوات المناسبة والضرورية للاختبار والتحليل، والوقت اللازم لدمج برمجية إكسبلويت محددة في عمليات قائمة أخرى، ورواتب الباحثين المشاركين في عملية تطوير برمجيات إكسبلويت، ومزيج قاعدة الرموز (أي احتمال الاضطرار إلى مراجعة برمجية إكسبلويت وتحديثها لتناسب نسخًا جديدًا من الرمز من أجل الحفاظ على قدرة ما)، والعرض والطلب على برمجية إكسبلويت لاستغلال منصة أو قاعدة رموز معينة.[27] وقد تأتي القيمة الإضافية من فرادة الثغرة (مثلًا إن كانت الثغرة الوحيدة المكتشفة في منتج محدد) أو حاجة العميل وإطاره الزمني.

وتتأتى الفوائد من تمتع برمجية إكسبلويت بأمد حياة كافٍ لتكون مفيدة للعميل. وبشكل مثالي، قد تبحث منظمة تطوير برمجيات إكسبلويت عن وسائل لخفض التكاليف وزيادة الفوائد.

تكاليف وفوائد اكتشاف ثغرات ثغرة يوم صفر وتطوير برمجيات إكسبلويت لها

إلى جانب تواريخ اكتشاف الثغرة للمرة الأولى بشكل خاص (تاريخ الميلاد) وتاريخ الإفصاح عنها علنًا (تاريخ الموت)، تتضمن بياناتنا تاريخ تطوير برمجية إكسبلويت الفعالة بالكامل (تاريخ النضوج). وبالتالي، راقبنا الوقت اللازم لتطوير برمجية إكسبلويت باعتباره الوقت الفاصل بين الميلاد والنضوج.[28] يظهر الشكل 3.9 التواتر الضروري لتطويربرمجية إكسبلويت لبياناتنا.[29]

لقد وجدنا أن الوقت اللازم لتطوير برمجية إكسبلويت يختلف، لكنه قصير نسبيًا بشكل عام. وفي بياناتنا، جرى تطوير 71% من برمجيات إكسبلويت خلال شهر (31 يومًا أو أقل) وحوالى الثلث (31.44%) في أسبوع أو أقل، في حين استغرقت نسبة 10% فقط أكثر من 90 يومًا.[30] وقد استغرقت معظم برمجيات إكسبلويت في مجموعة بياناتنا بين 6 و37 يومًا لتصبح فعالة بالكامل (بمتوسط 22 يومًا، مع حد أدنى يبلغ يومًا واحدًا وحد أقصى يبلغ 955 يومًا).

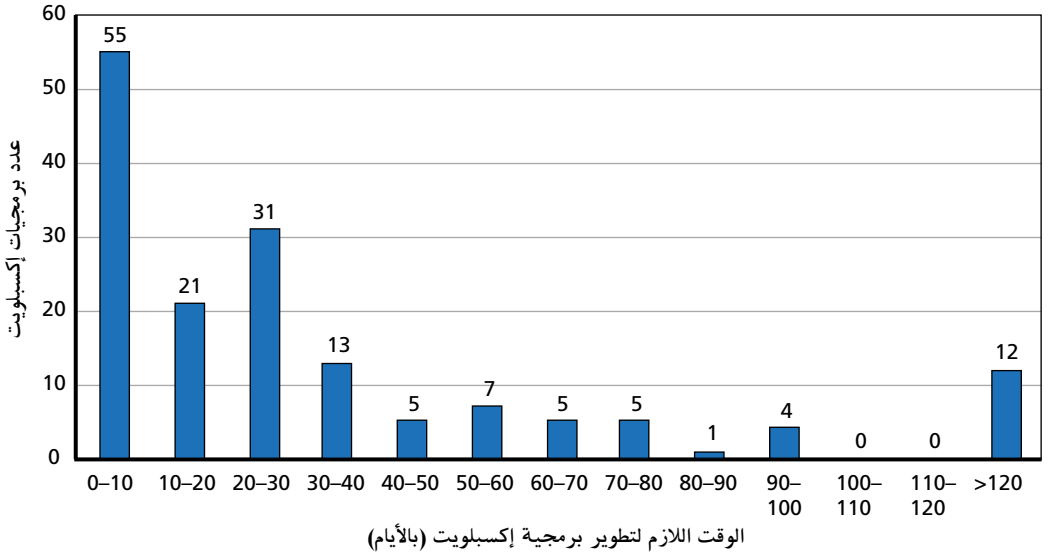
[27] مثلًا يقال إن مكتب التحقيقات الفيدرالي أنفق حوالى مليون دولار على التقنية (التي يشكك الكثيرون في أنها كانت ثغرة يوم صفر) المستخدمة لفتح جهاز آي فون (iPhone) هوسنبول Hosenball، 2016. وكان الطلب المرتفع على فتح الجهاز سببًا في ارتفاع التكلفة.

[28] لم تشمل بياناتنا الوقت الذي استغرقه اكتشاف ثغرة للمرة الأولى (وقت البحث) رغم أننا تحدثنا إلى عدة أشخاص شاركوا في عملية البحث عن الثغرة وتطوير برمجية إكسبلويت. وتعلمنا أن الوقت اللازم لاكتشاف ثغرة أطول عادة من الوقت الضروري لتطوير برمجية إكسبلويت فعالة بالكامل. وبالتالي، فور اكتشاف ثغرة، قد يتم تطوير برمجية إكسبلويت فعالة بالكامل بشكل سريع نسبيًا. وبات من المسلم به أنه على الفرد ضمان وجود البنية التحتية الضرورية للاختبار وأن التأسيس قد يستغرق وقتًا أطول. للمزيد من التفاصيل، انظر الملحق E، "شراء برمجية إكسبلويت لاستغلال ثغرة يوم صفر: بعض التكاليف والاعتبارات السعرية".

[29] لم تظهر أي اتجاهات بارزة لبرمجيات إكسبلويت هذه (n = 16) التي استغرقت عملية تطويرها أكثر من 90 يومًا. فأكثر من 56% منها لا تزال مجهولة (واحدة خالدة وأخرى تحتوي على عامل إعادة هيكلة الرمز)، وحوالى 31% معلومة ولديها إصلاحات أمنية. في حين شكلت نسبة 56% ثغرات فساد الذاكرة (الانقسام بين التدفقات المفرطة المكونة والمكدسة). وعلى صعيد نوع برمجيةإكسبلويت، كانت 50% منها من النوع البعيد، و25% محلية، ونحو 18.75% من جانب العميل و6% تابعة. كما استقرت 69% منها تقريبًا في منصة ويندوز. يذكر أنه تم اكتشاف الثغرات بين 2005 و2014، حيث اكتشفت الأغلبية في 2011.

[30] استغرقت نسبة 3% تقريبًا أكثر من عام، حيث استغرق تطوير برمجية إكسبلويت لاثنتين من الثغرات بين عامين وعامين ونصف. وقد تم استغلال الأولى في 2007 وهي تتمتع بحالة غير مؤكدة. أما الثانية فجرى استغلالها في 2010 وهي خالدة بما أنها في قاعدة رموز لا تتم صيانتها. وقد تضمنت الحالتان برمجية إكسبلويت عن بعد باستخدام ثغرات فساد الذاكرة (تدفق مفرط) ضمن ويندوز. وقد عزي الوقت الطويل لتطوير برمجية إكسبلويت إلى مشاكل على صعيد الموثوقية.

الشكل 3.9
تواتر فترة تطوير برمجية إكسبلويت (n=159)



RAND *RR1751-3.9*

وقد يعزى وقت تطوير برمجيات إكسبلويت السريع نسبيًا، بشكل جزئي إلى إقدام الباحثين عن الثغرات أنفسهم باستغلال النوع نفسه من الثغرات أو المنصات. فعلى سبيل المثال، سيعرف على الأرجح شخص خبير في استغلال نظام آي أو إس (iOS) من آبل (Apple) المبادئ الأساسية التي يجب اعتمادها (وحدات بناء برمجية إكسبلويت)، لذا بعد اكتشاف ثغرة ما، يمكنه إتمام الاستغلال بشكل أسرع من شخص يعمل حديثًا على هذا النظام.

ويمكن أن يعزى إطار زمني طويل للاستغلال إلى عدة أسباب: (1) مشاكل في الموثوقية، كما كانت عليه الحال في الأمثلة التي ذكرناها سابقًا،[31] و(2) الحاجة إلى ربط عدة ثغرات ببعضها البعض لاستحداث برمجية إكسبلويت، و(3) طلب العميل أن تعمل برمجية إكسبلويت على عدة نسخ، و(4) اعتبار ثغرة غير مهمة مؤقتًا ووضعها جانبًا.

هذا ولا تعني فترة استغلال طويلة بالضرورة أن أحدًا ما كان يعمل على تطوير برمجية إكسبلويت طيلة الوقت. في الواقع، قد يكون محلل ما قد اكتشف الثغرة واستثمر بعض الوقت في محاولة استغلالها، وفي حال لم يتم استغلالها فورًا أو بسهولة في غضون شهر أو إطار زمني معين، فيجب تمريرها إلى مطور آخر أو "وضعها جانبًا" لعدة أشهر إلى حين إصلاح الخطأ المانع أو إلى أن يزداد اهتمام العميل بالثغرة (مثلًا، عندما يطلب عميل تطوير برمجية إكسبلويت لمنتج معين، بغض النظر عن الوقت المطلوب).

[31] يعود الأمر إلى أن برمجية إكسبلويت كانت فعالة في المختبر ولكن ليس على أرض الواقع.

لقد قمنا أيضًا بتقييم فترة تطوير برمجية إكسبلويت بالمقارنة مع أمد الحياة (من النضوج إلى الموت) ووفق مختلف الخصائص. وتمامًا كما هو الحال في استنتاجنا القائل بأنه ما من خاصية في برمجية إكسبلويت تشير إلى طول أو قصر حياتها من حيث الأهمية الإحصائية، وجدنا أن الوقت اللازم لتطوير برمجية إكسبلويت لم يؤثر على فترة حياة البرمجية ولم يتسم بميزة الإشارة إلى حياة طويلة أو قصيرة لتطوير برمجية إكسبلويت.[32]

للمزيد من التفاصيل حول التحليل الأولي لتكلفة تطوير برمجية إكسبلويت، لنظر الملحق E، "شراء برمجية إكسبلويت لاستغلال ثغرة يوم صفر: بعض التكاليف والاعتبارات السعرية".

[32] تمامًا كتحليل الصمود، فحصنا نوع الثغرة والمنصة المتضررة ونوع المصدر وفئة برمجية إكسبلويت، فضلًا عن الحالة.

استنتاجات وتداعيات

تتمثل قيمة التحقيق في ثغرات يوم الصفر في أن الاستنتاجات الناجحة أو الأسئلة التي بقيت بدون إجابة يمكنها المساعدة على الإشارة إلى المناطق الأمثل من البحث للمدافعين والأطراف التي تركز على الهجوم.

وقد أسفر بحثنا عن عدة استنتاجات سنسلط الضوء على اثنين منهما:

1. تتمتع برمجيات إكسبلويت وثغراتها الأساسية بمتوسط أمد حياة متوقع طويل (6.9 سنوات).
2. بالنسبة إلى مخزون محدد من ثغرات ثغرة يوم صفر، تمّ بعد عام واحد، اكتشاف حوالى 5.7% منها من قبل جهة خارجية.

وتشير هذه الاستنتاجات إلى أن التخزين قد يكون ذكيًا بالنسبة إلى الذين يركزون على الهجوم، ويرغب الباحثون عن الثغرات الذين يعملون بتقنيات متطورة في تخزين الثغرات التي يكتشفونها، عوضًا عن الإفصاح عنها. أما المدافعون فسيبقون دائمًا عرضة لثغرات ثغرة يوم الصفر، وسيرغبون دائمًا على الأرجح بالإفصاح عن الثغرة وإصلاحها فور اكتشافها. سنقوم في ما يلي بمناقشة جميع استنتاجاتنا وتداعياتها على الدفاع والهجوم.

الاستنتاج 1: قد يكون إعلان ثغرة حية (مجهولة للعامة) أو ميتة (معلومة للعامة) مضللًا أو مبسطًا للغاية

تكمن الممارسة الشائعة في تصنيف الثغرة ببساطة باعتبارها حية (مجهولة للعامة) أو ميتة (معلومة للعامة). غير أن تحليلنا كشف عن وجود عدة فئات ثانوية لكل ثغرة، ما قد يجعل توصيف الثغرة باعتبارها حية أو ميتة مضللًا ومبسطًا للغاية. ولقد وجدنا أن الثغرات قد تكون في حالة شبه ميتة بسبب عامل إعادة هيكلة الرمز، وأنها قد تموت لعدة أسباب، وأنها ديناميكية، أي ما يمكن استغلاله اليوم قد لا يمكن استغلاله في اليوم التالي (والعكس صحيح).

كما أنه خلال التحقيق في حالة الثغرات، وجدنا أن أنظمة الثغرات الشائعة والانكشاف (CVE) لا توفّر دائمًا معلومات كاملة ودقيقة عن مدى خطورة الثغرات.

تداعيات الدفاع

قد تتسبب المبالغة في تبسيط الثغرات باعتبارها حية (مجهولة للعامة) أو ميتة (معلومة للعامة) في بروز عائق أمام مساعي اكتشاف الثغرات.

فقد تتضمن منتجات "مرحلة الاحتضار" ثغرات خامدة (أي خالدة)، لذا على البائعين والمدافعين بذل الجهود الضرورية لوضع حد للبرمجيات والتشجيع على تحديث البرمجيات إلى نسخها الجديدة. ويجب بذل جهود مماثلة للبحث عن نسخ سابقة لقواعد الرموز، سواء كانت غير مدعومة أو لا، قد يكون احتمال مواصلة استخدامها كبيرًا (مثلًا في أنظمة المراقبة الصناعية) في الثغرات. وبما أن قواعد رموز مرحلة الاحتضار تبقى ثابتة نسبيًا، قد تقلص كل ثغرة مكتشفة العدد الإجمالي للأخطاء.

تداعيات الهجوم

في حين قد يكون استعمالها محدودًا، قد تكون قيمة الثغرات الخالدة أو التي تضم عامل إعادة هيكلة الرمز كبيرة بالنسبة إلى العمليات، رهنًا بالنظام أو الهدف الكامن فيها، وعلى الباحثين التفكير باستمرار في إعادة النظر في الثغرات التي اعتبروا يومًا ما أنها غير قابلة للاستغلال.

الاستنتاج 2: إن متوسط أمد الحياة المتوقع لبرمجيات إكسبلويت يبلغ 6.9 سنوات بعد اكتشافها الأولي، لكن حوالى 25% من برمجيات إكسبلويت لن تصمد لأكثر من عام ونصف في حين أن نسبة 25% أخرى ستصمد لأكثر من 9.5 سنوات

بعد الاكتشاف الأول من قبل باحث ثغرات، يصب متوسط أمد الحياة المتوقع لبرمجيات إكسبلويت إلى 6.9 سنوات (تحديدًا 2521 يومًا)، ونقدر بنسبة ثقة تبلغ 95% أن تدوم أي برمجية إكسبلويت بين 5.39 و8.84 سنوات، علمًا بأن 25% فقط من الثغرات لا تصمد إلى 1.51 سنة، في حين تستمر 25% فقط أكثر من 9.5 سنوات. ويعني أمد الحياة الطويل المتوقع نسبيًا عند 6.9 سنوات أن ثغرات يوم الصفر، لا سيما تلك التي تُعدّ برمجيات إكسبلويت التابعة بها للاستعمال الخاص (أي في الأسواق الرمادية)، قديمة على الأرجح.

وفي حين أظهرت بياناتنا أن "الحياة القصيرة" للثغرة تبلغ 1.5 سنة، قد تكون هذه الفترة طويلة بما يكفي بالنسبة إلى معظم باحثي الثغرات. وقد أبلغتنا إحدى فرق تطوير برمجيات إكسبلويت أنها تفضل عادة موت الثغرات بعد 18 شهرًا (من خلال مزيج الرموز وليس من خلال إعادة اكتشاف مستقلة) من أجل مواصلة مزاولة أعمال بيع برمجيات إكسبلويت جديدة. كما أشار الفريق إلى أنه، بحسب تجربته، كان العملاء مسرورين عند بقاء برمجية إكسبلويت قابلة للنمو وبقاء ثغرة يوم الصفر لمدة 12 شهرًا على الأقل، رغم أن ذلك اعتُبر معيارًا عاطفيًا أكثر منه قائمًا على البيانات التشغيلية.

تداعيات الدفاع

يجب أن يتوقع المدافعون أن تكون أي ثغرات ثغرة يوم صفر في رمزهم تشكل فائدة كبرمجيات إكسبلويت على وجه الخصوص مدفونة بشكل عميق على الأرجح في قواعد الرموز. قد يعني ذلك أنه على البائعين مواصلة البحث عن الثغرات حتى في مجالات الرمز التي راجعوها بدقة. وقد يساهم وجود المزيد من الأشخاص الذين يساعدون على اكتشاف الثغرات من خلال برامج المكافأة الأمنية في حث باحثين جدد باستمرار على دراسة رمز يبدو أساسًا نظيفًا.

ويصبح إيجاد الثغرات في الرمز وإصلاحها أصعب عند إعادة استعمال البرمجية بين المنصات، ولا سيما مع إنترنت الأشياء (Internet of Things)، حيث يتمّ استعمال جزء أكبر من الرمز في مختلف المنصات ضمن البيئة المنزلية والنقل والأنظمة الطبية وغيرها. وغالبًا ما تعني طبيعة أجهزة إنترنت الأشياء عدم وجود آليات لإصلاح الجهاز، كما يطرح اكتشاف الثغرات وإصلاحها تحديًا ما لم تبذل جهود حثيثة للحفاظ على البنية التحتية من أجل إصلاحها.

هذا وقد كون "فساد البرمجية" (Software rot)، الذي يشير إلى ما يحصل عندما لا يتم تحديث البرمجية بانتظام ولا تتم صيانة الرمز، مصدر قلق بسبب وجود ثغرات خامدة سيجري اكتشافها مع مرور الوقت وقد تبقى مفيدة لاستغلالها. وعليه، قد يكون من المفيد بذل جهود لإصلاح ثغرات يوم الصفر في برمجية لا تجري صيانتها، لا سيما بالنسبة إلى الرمز الذي يستخدم على نطاق واسع. (على وجه الخصوص بالنسبة إلى الرمز الذي أنتجته شركات البرمجيات الزائلة، لا يبرز فورًا خيار إصلاح أو إطلاق برنامج المكافأة الأمنية). ويمكن تطبيق هذا النوع من الجهود أيضًا على برمجيات مفتوحة المصدر.

ويشير أمد حياة متوقع عند 6.9 سنوات إلى أن الهجوم قد يكون مهيمنًا. وقد صح ظننا باعتبار أن المدافعين يحتاجون على الأرجح إلى أدوات أفضل وإلى المزيد من الخيارات لاكتشاف ثغرات ثغرة يوم صفر ورصدها عندما يتم استغلال نظام أو حزمة برمجيات. كما أنه عوضًا عن التركيز فقط على اكتشاف ثغرات يوم الصفر أو استحداث المزيد من الأدوات والخيارات التي تتيح عملية الرصد عند استغلال نظام أو حزمة برمجيات، قد يتمكن المدافعون من ترجيح الكفة لصالحهم عبر البدء من فرضية المساومة، والتحقيق في سبل تحسين تصميم هندسة النظام من أجل احتواء أثر المساومة.

تداعيات الهجوم

يجب أن يدرك مطورو برمجيات إكسبلويت أو أولئك الذين يحتفظون بثغرات يوم الصفر أن 25% من برمجيات إكسبلويت التابعة لهم لن تبقى مفيدة (كثغرات يوم صفر) في غضون عام ونصف. وفي الوقت نفسه، بما أن الثغرة المتوسطة تبقى ثغرة يوم صفر لنحو 7 سنوات، قد يتراجع الطلب على الثغرات في حزمة برمجيات معينة مع مرور الوقت. وقد لا تحتاج جهات الهجوم إلى تخزين الكثير من برمجيات إكسبلويت بعمق لحزمة برمجيات معينة، نظرًا إلى حياة الثغرات الطويلة. فالاحتفاظ بعدد قليل من الثغرات الاحتياطية قد يكون كافيًا. ويمكن إرسال أي ثغرات مماثلة متبقية إلى البائع لإصلاحها، بما أنها لن تكشف الثغرة التي من المزمع أن تبقى

جاهزة للعمل.[1] وقد يكون من المفيد تطوير "شعاع توقيف" (cut-off radius) على مقربة من الثغرة يجب أن يبقى حيًا.

أما الذين يشاركون في التخطيط للعمليات الهجومية باستخدام ثغرة يوم صفر محددة فعليهم استخدامها فقط في ظروف التخطيط القصيرة الأمد.

من ناحية أخرى، قد يسمح استعمال أي ثغرة يوم صفر بإطار زمني أطول للتخطيط أو تنفيذ عملية.

الاستنتاج 3: لم تشر خصائص ثغرة ما إلى حياة طويلة أو قصيرة، لكن قد تسعى تحليلات مستقبلية إلى مقارنة فيروس لينكس (Linux) مع أنواع أخرى من المنصات، ومراقبة أوجه الشبه بين رمز مصدر مفتوح وآخر مغلق وتجمعات مختلفة لنوع فئة برمجية إكسبلويت

بعد تقييم نوع الثغرة والمنصة المتضررة ونوع رمز المصدر ونوع فئة برمجية إكسبلويت، لم تظهر أي خاصية من الناحية الإحصائية باعتبارها "دليلًا" قد يشير إلى حياة قصيرة أو طويلة. وقد يُعزى ذلك سواء إلى نقص فعلي في الترابط أو نقص في القوة الإحصائية لرصد تلك الترابطات، نظرًا إلى العدد الصغير نسبيًا من الثغرات الميتة في مجموعة بياناتنا. وتمثل العائق الأكبر في غياب حالات الموت المسجلة ضمن مستوى كل خاصية، ما حدّ من القوة الإحصائية على رصد الفوارق بين مختلف المستويات. وبالتالي، قد توفر المزيد من البيانات عددًا أكبر من النتائج المهمة إحصائيًا، بغض النظر ما إذا كان ذلك سيؤكد هذه النتيجة أو يجد أن خاصية محددة ومهمة غير واضحة.[2]

وفي حين لم يكتسب أي أمر أهمية إحصائية، يوفر تحليلنا توجيهات بشأن أي فرضيات قد يكون من المفيد تجربتها في تحليلات مستقبلية، وبخاصة لمراقبة ديمومة ثغرات لينكس بالمقارنة مع منصات أخرى، ولتأكيد أوجه شبه ديمومة الثغرات الخاصة بنوع الرمز المفتوح والمغلق المصدر، وللتحقيق في أهمية جمع برمجيات إكسبلويت من جانب العميل وبرمجيات إكسبلويت عن بعد معًا مقارنة بجمع برمجيات إكسبلويت محلية ومختلطة وغير ذلك.

ولو ظهرت خصائص مهمة من الناحية الإحصائية (يمكن تأكيدها أو دحضها بمزيد من البيانات)، لساعد ذلك أولئك المنخرطين في البحث عن الثغرات أو عملية معالجة الثغرات أو المجتمع الأمني عمومًا على صقل القرارات المتعلقة بما يجب الاحتفاظ به وما يجب نشره بشكل علني. في حال اكتشاف نوع واحد من الثغرات أكثر من سواه خلال فترة قصيرة (مثلًا تدفقات المفرطة المكدسة لماك أو.إس OSX)، يجب ربما الكشف علنًا عن هذا النوع من الثغرات وإصلاحه (أي عدم تخزينه) لأن الوقت المتاح أمام الراغبين في استغلالها لعملياتهم

[1] رغم أنه، كما سنرى أدناه، غالبًا ما يؤدي اكتشاف ثغرة والإفصاح عنها إلى اكتشاف ثغرات مماثلة أخرى، قد يشكل ذلك عاملًا يساهم في زيادة الاصطدامات.

[2] لقد قدر أحد المدراء السابقين لمطوري برمجيات إكسبلويت أنه لن تظهر قط أي خاصية قد تشير إلى حياة طويلة أو قصيرة. عوضًا عن ذلك، اعتقد أن المؤشر الأفضل على الحياة الطويلة كان اكتشاف ثغرة في فئة سبق لباحثين آخرين أن دققوا فيها (مثلًا أخطاء في نوع الخط). وقد وافق باحثو ثغرات آخرون تحدثنا إليهم على هذه الفرضية.

الشخصية قد يكون قصيرًا، وقد يصبح عدد أكبر من الأفراد بخطر بسبب معدل الاكتشاف السريع.[3] وعلى نحو مماثل، فإذا كان نوع أو جانب معين من الثغرة يشير إلى فترة طويلة، قد يكون التخزين خيارًا مبررًا.

تداعيات الدفاع

يجب أن تركز الاستراتيجيات الأمنية على كافة أنواع الثغرات عوضًا عن نوع واحد فقط (مثلًا ثغرات فساد الذاكرة فقط، أو تلك الموجودة في منتجات مفتوحة المصدر).لا يمكن أن يركز المدافعون على شيء واحد في ظل غياب أي خاصية تجعل الثغرات أكثر أو أقل ترجيحًا.

تداعيات الهجوم

بما أنه لا يبدو أن هناك خاصية في الثغرات قد تشير إلى حياة أطول أو أقصر، قد يكون تخزين برمجيات إكسبلويت وتطويرها أكثر فعالية وكفاءة من حيث التكلفة لكافة الثغرات التي يسهل اكتشافها أو الأكثر تأثيرًا. وكما يظهر فإن بياناتنا لا تشير إلى وجود ثغرات "أقوى" أو "أضعف" من سواها من حيث مقاومة اكتشافها والإفصاح عنها.

الاستنتاج 4: اكتشفت أطراف أخرى نحو 5.7% من الثغرات لمجموعة محددة من ثغرات يوم الصفر بعد مرور سنة وأفصحت عنها

يُعتبر توقيت "التخلص" من مجموعة ثغرات ميتة مهمًا. فقد تغيّرت معدلات الاصطدام والتداخل بشكل كبير وفق الفاصل الزمني المستخدم (من 40% إلى أقل من 1%).[4] وقد وجدنا قيمة تداخل متوسطة تبلغ 5.76% (انحراف معياري بنسبة 6.79%) مقابل فاصل زمني لمدة 365 يومًا، وقيمة تداخل متوسطة تبلغ 0.87% (انحراف معياري بنسبة 5.3%) مقابل فاصل زمني لمدة 90 يومًا. وقد أسفر فاصل زمني لمدة 14 عامًا (أي كافة بياناتنا خلال فاصل زمني واحد) عن تداخل بنسبة 40%.

وباستثناء الفاصل الزمني لمدة 14 عامًا، أظهرت بياناتنا معدل اصطدام منخفض نسبيًا. وقد يُعزى ذلك إلى أن الأطراف ضمن مجال الاستغلال الخاص والأسواق الرمادية يتطلعون إلى ثغرات مختلفة عن تلك التي يبحث عنها صائدو الثغرات في المنشورات العامة ومجالات السوق الأبيض، فضلًا عن استخدام تقنيات مختلفة لإيجاد الثغرات، ما يسفر عن اكتشاف ثغرات مختلفة (على سبيل المثال الثغرات المكتشفة من خلال اختبار البرمجيات الآلي أو "الاختبار العشوائي" مقابل تلك المكتشفة من خلال تحليل يدوي).

[3] حتمًا، يتوقف الأمر على المنصة والنظام المستهدف المتضرر. وحتى إن كان يمكن تطوير برمجية إكسبلويت لاستغلال ثغرة بسرعة وهي تتمتع بأمد حياة قصير، قد يكون استخدامها لفترة قصيرة مفيدًا.

[4] إنه مثل ممتاز على كيفية استخدام البيانات لدعم وجهات نظر مختلفة، لا سيما معدل الاصطدام المرتفع مقابل المنخفض. وبالنسبة إلى الذين قالوا إن التداخل صغير (أو كبير)، فهم لا يحددون الفاصل الزمني بشكل متطابق. لقد أثبتنا من خلال بياناتنا أن النطاق قد يتراوح بين جزء من النسبة المئوية إلى نحو النصف، بحسب الإطار الزمني. ويسلط تحليلنا الضوء على أهمية تحديد الفاصل الزمني، وإلا يمكن للجميع على ما يبدو "إثبات" النتائج التي يريدون إظهارها، حتى مع وجود بيانات.

هذا وقد قمنا بمقارنة معدلات التداخل بين الذين يستهدفون الاستعمال الخاص (الدائرة الزرقاء في الشكل 1.2) وأولئك الذين يسعون إلى النشر العام (الدائرة الخضراء في الشكل 1.2). وقد يولي المسؤولون عن اعتبارات التخزين أهمية أكبر للتداخل بين المجموعتين المتعارضتين (الدائرتين الزرقاء والحمراء في الشكل 1.2)، علمًا بأنه قد يختلف عما خلصت إليه بياناتنا. فمن جهة، قد يكون معدل الاصطدام أعلى لأن هذا النوع من الباحثين يبحث ربما عن ثغرات مشابهة أو عبر استخدام تقنيات مشابهة. لكن قد يكون أكثر انخفاضًا لأن الذين يعنون بالاستغلال الخاص لا يميلون إلى التخزين العميق للهدف نفسه (أي طالما أن الثغرات قليلة ضمن الهدف المحدد، لا حاجة عادة إلى منح الأولوية لاكتشاف المزيد من الثغرات في ذلك الهدف). في هذه المرحلة، زعم فريق يعنى بالبحث عن الثغرات كنا قد تحدثنا إليه أن المشاركة في اكتشاف الثغرات تحدث أقل من مرة واحدة سنويًا.

ويتفق الكثير من باحثي الثغرات على أنه حيثما توجد ثغرة، توجد حتمًا ثغرات أخرى. وعليه، فور اكتشاف ثغرة (والإفصاح عنها علنًا) في جزء من الرمز، يغزو صائدو الأخطاء ومطورو برمجيات إكسبلويت هذا الجزء للبحث عن ثغرات أخرى.[5] وفي حالات نادرة تمكن فريق تطوير برمجيات إكسبلويت من الحفاظ على قدرات في برمجيات سعت فرق أخرى تبحث عن الثغرات وتتمتع بتمويل كبير إليها لسنوات.[6]

في حال موت نسبة كبيرة من برمجيات إكسبلويت، عندها قد يشير التداخل الكبير بين ما هو مكتشف بشكل علني وخاص إلى وجود تداخل أكبر حتى بين ما وجدته مختلف المجموعات التي تبحث سرًا عن الثغرات وتطوير برمجيات إكسبلويت لاستعمالها الخاص. انطلاقًا من هذا، قد تدل النسبة المئوية الصغيرة لبرمجيات إكسبلويت الميتة إلى أن الذين يبحثون عن الثغرات بشكل خاص لديهم نسبة تداخل صغيرة فيما بينهم.

تداعيات الدفاع

قد لا يكون اكتشاف الثغرات والاستعانة بمصادر خارجية كافيًا لضمان أمن الأنظمة، لأن التداخل بين ما هو مكشوف ومفصح عنه علنًا وما هو مكتشف سرًا يكون صغيرًا نسبيًا على ما يبدو. ويشير ذلك إلى أن الثغرات إما تكون كثيفة أو يصعب اكتشافها.

يدل وجود أي تداخل على أن عدة أشخاص بإمكانهم اكتشاف، أو سبق لهم أن اكتشفوا، الثغرات نفسها في منتجات تجارية، لا سيما تلك الثغرات المهمة للجهات الحكومية الفاعلة. بناء على ذلك، في حين تعتبر مقاربات على غرار المكافآت الأمنية ورصد الثغرات وإصلاحها قيّمة، على المدافعين التفكير في احتمال بذل المزيد من الجهود لاستحداث عمليات تخفيف، أو استخدام تدابير دفاعية أفضل، أو إجراء مراجعة أكثر دقة (أعمق) للرمز قبل طرح المنتج

[5] ثمة العديد من التشبيهات لذهنية الغزو هذه السائدة بين الباحثين عن الثغرات. وقد شملت عبارات رددها الناس ما يلي "عندما تجد خطأ ما ستجد آخر"، و"الأمر أشبه بكرة قدم الأطفال، يركض أحد خلف الكرة ويتبعه الآخرون"، "وعلى الرغم من أن بحر الأخطاء عميق وواسع، بإمكان الباحثين عن الثغرات رصد قطرة دم واحدة في المياه" و"الافتراس".

[6] في حين يكون مجتمع الباحثين عن الثغرات وأمن المعلومات مفتوحًا عمومًا وراغبًا في مشاركة المعلومات، تكون حتى الإشارة إلى الباحث عن الثغرات محدودة. فبالنسبة إلى مطوري برمجيات إكسبلويت المتمرسين، يعتبر تلميح شخص ما إلى وجود خطأ في مكتبة معينة أو جزء من الرمز كافيًا للمطورين كي يجدوا الخطأ. وعليه، يتكتم الباحثون الأمنيون أحيانًا عن أنواع الثغرات التي اكتشفوها. ويعتقد الكثيرون أنه حتى إن ألمحوا أو تحدثوا بشكل فرضي عن ثغرة في قاعدة رموز معينة، سيتدفق الباحثون الآخرون إليها لاكتشاف نوع مطابقة أو مشابه للثغرة.

للعامة، أو دراسة التقنيات المرتبطة بكيفية اكتشاف الهجمات للثغرات. علاوةً على ذلك، يمكن للمدافعين معالجة مسألة أمن البرمجيات من خلال اعتبار جميع الرموز غير آمنة، وبذل الجهود لتصميم هندسة دقيقة للشبكة والنظام لاحتواء أثر أي مساومة محتملة.

تداعيات الهجوم

قد يستخدم باحثو الثغرات الذين يركزون على الهجوم أساليب مختلفة لاكتشاف الثغرات من أولئك الذين يركزون على الدفاع. وبالتالي، قد يستفيد معسكر الهجوم من مواصلة اكتشاف الثغرات بفضل التقنيات الحالية التي يستخدمونها. وقد يعني هذا أيضًا أن إضافة باحثي ثغرات إلى الفريق قد يساعد على إيجاد ثغرات جديدة وفريدة.

ورغم أن بياناتنا كشفت نسبة تداخل محدودة نسبيًا، فإن هذه النسبة برزت فقط بين الذين يسعون إلى الاستعمال الشخصي والذين يسعون إلى النشر العام (أي التداخل بين الدائرتين الزرقاء والخضراء في الشكل 1.2)، عوضًا عن مجموعتين متنافستين تبحثان عن ثغرات للاستعمال الخاص (أي الدائرتين الزرقاء والحمراء في الشكل 1.2). وسيتم صقل استنتاجاتنا بشكل إضافي من خلال تحسين المعلومات حول تواتر التقييم (أي الفاصل الزمني الذي تستخدمه المنظمات والوكالات)، إلى جانب نوع الثغرات التي تحتفظ بها مجموعات خاصة أخرى.

الاستنتاج 5: فور اكتشاف ثغرة يمكن استغلالها، يكون الوقت الضروري لتطوير برمجية إكسبلويت فعالة بالكامل سريعًا نسبيًا، حيث يبلغ متوسطه 22 يومًا

وقد لاحظنا تفاوت مدة تطوير برمجيات إكسبلويت، لكنها بوجه عام تكون قصيرة نسبيًا. وأظهرت بياناتنا أن 71% من برمجيات إكسبلويت تمّ تطويرها في غضون شهر واحد (31 يومًا أو أقل)، وحوالى الثلث (31.44 %) في غضون أسبوع أو أقل، في حين استغرق تطوير 10% منها فقط أكثر من 90 يومًا. واستغرقت أغلبية برمجيات إكسبلويت في مجموعة بياناتنا بين 6 و37 يومًا لتصبح فعالة بالكامل (بمتوسط 22 يومًا، مع حدّ أدنى يبلغ يومًا واحدًا وحدّ أقصى يبلغ 955 يومًا).

وقد تتوقّف تكلفة تطوير (وبالتالي قيمة أو سعر) برمجيات إكسبلويت على عدة عوامل تشمل: الوقت الضروري لاكتشاف ثغرة يوم صفر قابلة للنمو (وقت البحث)، والوقت اللازم لتطوير برمجية إكسبلويت للاستفادة من ثغرة يوم صفر (وقت تطوير برمجية إكسبلويت)، وتكلفة شراء جهاز أو رمز للمراجعة أو الاستحواذ عليه، والوقت الضروري لإعداد مختبر تجارب وتكلفة شراء البنية التحتية أو الأدوات المناسبة والضرورية للاختبار والتحليل، والوقت اللازم لدمج برمجية إكسبلويت محددة في عمليات قائمة أخرى، ورواتب الباحثين المشاركين في عملية تطوير برمجيات إكسبلويت، ومزيج قاعدة الرموز، أي احتمال الاضطرار إلى مراجعة برمجية إكسبلويت وتحديثها لتناسب نسخًا جديدة من الرمز من أجل الحفاظ على قدرة ما، والفترة التي تكون خلالها برمجية إكسبلويت جاهزة للاستعمال لكن دون استعمالها، والعرض والطلب على برمجية إكسبلويت لمنصة أو قاعدة رموز معينة. وقد تأتي القيمة

الإضافية من فرادة ثغرة (مثلًا إن كانت الثغرة الوحيدة المكتشفة في منتج محدد) أو حاجة العميل وإطاره الزمني.

وبحسب الوقت الضروري لاكتشاف ثغرة، ونظرًا إلى الموهبة الصحيحة وبعد اختبار البنية التحتية، قد يكون تطوير برمجيات إكسبلويت محليًا بدلًا من الاستعانة بمصادر خارجية أقل تكلفة. وبالاستناد إلى بيانات دقيقة حول الوقت اللازم لاكتشاف ثغرة، يمكن إصدار تقديرات لمساعدة أولئك الذين يشترون ثغرات على معرفة السعر الذي يجب أن يدفعوه (وفق تكاليف العمالة وحدها). حتمًا، لا يمكن أن ننسى دور خطورة الثغرة وأثرها. وفي حين بلغ متوسط الفترة الضرورية لتطوير برمجية إكسبلويت 22 يومًا، وصل الحد الأقصى إلى سنتين ونصف، ما يشير إلى أن عمليات تطوير برمجيات إكسبلويت تتطابق وغالبًا ما تتطلب باحثين موهوبين ومتفانين كثيرًا.

والمثير للاهتمام أن وقت تطوير برمجية إكسبلويت لا يؤثّر على أمد حياة أو استمرارية هذه البرمجية.

تداعيات الدفاع

لا تأتي غالبية تكلفة برمجيات إكسبلويت لاستغلال ثغرة يوم صفر من العمالة بل من القيمة المتجذرة فيها وغياب العرض. ولا يجدر ببائعي برمجيات السوق الأبيض وبرامج المكافأة الأمنيةدفع الأموال لقاء الثغرات استنادًا إلى العمالة وحدها، بما أنهم لا ينافسون جهات أخرى على شراء برمجيات إكسبلويت لاستغلال ثغرات يوم الصفر للاستعمال الخاص.

تداعيات الهجوم

ببساطة، يمكن لأي مهاجم خطير الحصول على ثغرة يوم صفر بشكل ميسور لأي هدف تقريبًا. لكن يمكن أن تتسبب تكاليف ملموسة أخرى (الاستحواذ على منتجات لإيجاد ثغرات فيها، وإنشاء بنية تحتية للاختبار، والاحتفاظ ببرمجية إكسبلويت وتحويلها لتعمل على عدة نسخ، واستئجار مكان عمل، الخ) وتكاليف غير ملموسة (قيمة الطلب المرتفع، ومنتج محدود العرض، الخ) بارتفاع الأسعار بشكل ملحوظ.

وضمن بياناتنا، كان للثغرات التي تم شراؤها من أطراف ثالثة أمد حياة أقصر (بلغ متوسط الحياة 1.4 سنة). ويمكن استخدام ذلك كحجة لاكتشاف الثغرات وتطوير برمجيات إكسبلويت محليًا عند الرغبة في حياة أطول.

وتساهم الحكومات في ارتفاع الطلب في هذا السوق، ولهذا السبب تتمتع بقوة شرائية كبيرة. في حال قامت حكومة بتنسيق جهودها واستعملت قوتها الشرائية، يمكنها التحكم أكثر بأسعار برمجيات إكسبلويت.

توصيات أخرى للدفاع

قد يخلف المنخرطون في الدفاع أثرًا أكبر على ثغرات يوم الصفر من أولئك المنخرطين في الجانب الهجومي.

فقد يرغب الدفاع في التركيز على التخفيف والاحتواء والمساءلة والحفاظ على بنية تحتية متينة للإصلاح، عوضًا عن اكتشاف الثغرات وإصلاحها.[7] وتُعتبر عمليات التخفيف (على غرار آلية تحديد الوصول وأداة تجربة التّخفيف المحسّنة من مايكروسوفت) وسائل لإلغاء الثغرات التي يسهل اكتشافها لكنها ليست عقبة مستحيلة.[8] وقد يبرز الاحتواء على شكل عزل، رغم أنه يجب أن يكون ملموسًا، فلا تدخل مراقبات الأجهزة الافتراضية والعزل الافتراضي في الحسبان.[9] أما المساءلة فتعني معرفة البرمجية الموجودة في كل جهاز، ومعرفة الأجهزة المتواجدة في شركة ما، ومعرفة مكان وجود الوسائط القابلة للإزالة. وتطرح هذه المسألة تحديًا أكبر مع بروز إنترنت الأشياء (راجع ليبيكي وأبلون وويب، [2015] لمزيد من المناقشة).

يجب أن يأخذ الدفاع في الحسبان مقاربات استراتيجية إضافة إلى المقاربات التكتيكية الأكثر تقليدية. وذلك إلى جانب محاولة فهم ما الذي تقوم به برمجية إكسبلويت على وجه التحديد، حيث قد يكون من المفيد معرفة أن المساومة أو المحاولة حتمية وتهدف إلى الحد من أثر برمجية إكسبلويت من خلال الاحتواء والمساءلة.

ويعتقد البعض أنه يجب ان تهدف استراتيجية الحكومة إلى استحداث تهيئات جديدة وجمع أساليب التفاف جديدة تحمي العامة، لكن بدون الكشف عن أي معلومات إلى العامة حول ماهية هذه الوسائل للحماية. وقد يكون من المفيد أن تشارك الحكومة بصراحة أكبر مع العامة فائدة عمليات التخفيف التي تطلقها، لا سيما إذا كانت تتعلق بثغرات البرمجيات.

توصيات أخرى للهجوم

لا تقوم الشركات عادة بالتدقيق في الثغرات بعد إطلاق رمز جديد (أي عامل إعادة هيكلة الرمز)، لكن ربما يجدر بها فعل ذلك. ففي حين أنها ترغب مثاليًا بأن يستخدم جميع العملاء النسخة الأحدث، لا يكون هذا هو الواقع دائمًا. فبعض الأنظمة (مثلًا نظام التحكم الصناعي (ICS) ونظام التحكم الإشرافي وتحصيل البيانات (SCADA)) تتطلب بقاء النسخ القديمة في مكانها لفترة أطول مما هو مستحب. وفي أحيان أخرى، يعني التحديث إلى النسخة الأحدث أو تطبيق أحدث الإصلاحات الأمنية إعاقة سير العمل ويكون باهظًا للغاية. وعليه، يجدر بالمجموعات والمنظمات التي تعنى بأعمال البحث عن الثغرات وتطوير برمجيات إكسبلويت إعادة النظر في قواعد الرموز القديمة والنسخ السابقة، علمًا بأن الثغرات القابلة للاستغلال قد تكون لا تزال قائمة والمنتجات الأقدم لا تزال مستخدمة حول العالم.

[7] لكن تعتبر بعض الشركات اكتشاف الثغرات وإصلاحها إستراتيجية قابلة للنمو. مثلًا، في شباط (فبراير) 2002، أغلقت شركة مايكروسوفت أبوابها وأوقفت أعمال التطوير الخاصة بالإصلاحات الأمنية لشهرين. انظر الفصل 3، "دورة حياة التطوير الأمني: الحياة في المحولات الرقمية" على موقع مايكروسوفت (مايكروسوفت، غير مؤرخ).

[8] لا تعتبر حلول مكافحة الفيروسات عمومًا عقبة على الإطلاق. كان لBUSBY سبعة برمجيات إكسبلويت لمنتجات مكافحة الفيروسات، حيث أن خمس برمجيات منها لا تزال حية (الأكبر بينها يبلغ 4.4 سنوات)، وتبقى اثنتين في حالة غير مؤكدة.

[9] كانت خمس برمجيات من برمجيات إكسبلويت التابعة لBUSBY مخصصة لمراقبات الأجهزة الافتراضية أو الآلات الافتراضية. وقد ماتت برمجيتان منها في حين قتل BUSBY إحداها عمدًا.

وبغية مكافحة برامج اختبار البرمجيات (fuzzers) وصائدي الأخطاء (bug hunters) الذين يبحثون عن الثغرات في مجال الاستعمال العام، قد يرغب مطورو برمجيات إكسبلويت الخاصون في التدقيق في أجزاء من قاعدة الرموز الأكثر تعقيدًا (أو اقتباسًا عن باحث عن الثغرات "الأكثر إزعاجًا").

هل يجدر إيلاء ثغرات يوم الصفر هذا القدر من الاهتمام؟

يشكل التصيد الاحتيالي (Spear-phishing) (أي إرسال الرسائل الإلكترونية التي يبدو أنها من مرسل معروف أو موثوق لدفع المتلقي إلى الكشف عن معلومات سرية) هجومًا كبيرًا لا يعتمد عادة على الاستفادة من الثغرات أو من ثغرة يوم الصفر أو غير ذلك. وتعتمد غالبية الهجمات على ثغرات معروفة بالفعل. لذا ليس واضحًا مدى التركيز الذي يجب صبه على ثغرات يوم الصفر وبرمجيات إكسبلويت. وفي ظل وجود الكثير من الحالات حيث لم يطبق المستخدمون إصلاحات على ثغرات معروفة، هل من المنطقي التركيز على ثغرات يوم الصفر فحسب؟[10] وإلى جانب الحالات القليلة التي تكون فيها ثغرة يوم الصفر السبيل الوحيد لدخول هدف صعب، يمكن الاعتبار أن ثغرات يوم الصفر لا تكتسي أهمية ضمن المخطط الكبير للأمن الإلكتروني. وعليه، قد تبرز عمليات يعتبر فيها دخول نظام أو هدف ضروريًا إلى حدّ كبير، وتكون ثغرات يوم الصفر وبرمجيات إكسبلويت الطريق الوحيد إلى ذلك.

هل يجب تخزين الثغرات أو لا؟

قد تقرر الحكومات إبقاء ثغرات يوم الصفر سرية، سواء لأغراض دفاعية (مثلًا اختبار الانتشار) أو للعمليات الهجومية. ويتطلب قرار التخزين أخذ عدة عوامل في الحسبان بدقة، بما في ذلك الثغرة بحدّ ذاتها واستخدامها وظروف استخدامها وخيارات أخرى قد تكون متاحة لتحقيق النتيجة المرجوة.

وتُظهر تحليلاتنا أن ثغرات يوم الصفر قد تتمتع بمتوسط حياة طويل ومعدلات اصطدام منخفضة. وقد يشير التداخل المحدود إلى أن الثغرات كثيفة (أي وجود ثغرة مختلفة أخرى عادة) أو يصعب اكتشافها (حيث أن هاتين الخاصيتين ليستا حصريتين بالضرورة). في حال وجود ثغرة أخرى عادةً، قد يُعتبر مستوى الحماية الذي يحصل عليه المستخدمون من الباحثين الذين كشفوا ثغرة ما متواضعًا، وقد يستنتج البعض أن تخزين ثغرات يوم الصفر قد يكون خيارًا منطقيًا. في حالة صعوبة اكتشاف ثغرات يوم الصفر، قد يدعم حينها الاحتمال الضعيف بأن يجد آخرون الثغرة نفسها حجة الاحتفاظ بمخزون.

[10] على سبيل المثال، بعد مرور ست سنوات على اكتشاف كونفيكر (Conficker) وإصلاحه، لا تزال حوالى مليون آلة متضررة (أصغري Asghari وسيري Cieri وفان إيتين van Eeten، 2015). وتشير أبحاث سابقة (ليبيكي وأبلون وويب، 2015) إلى أنه لا يمكن حماية المنظمات الكبيرة بما يكفي من الاختراق، سواء بثغرات يوم الصفر أو ثغرات لا تمت لثغرة يوم الصفر بصلة.

من ناحية أخرى، يُظهر تحليلنا أن معدلات الاصطدام لثغرات يوم الصفر ليست معدومة. وقد يشير البعض إلى أنه في حال بروز أي احتمال أن يكتشف شخص آخر (بخاصة منافس) ثغرة يوم الصفر نفسها، يتم عندها تبرير التداعيات الخطيرة المحتملة لإبقاء ثغرة يوم صفر سريًا وترك العامة معرضين لخطر الكشف الفوري عن الثغرة وإصلاحها. ومن هذا المنطلق، قد يتمثّل القرار الأفضل بالتخزين فقط في حال كان المرء واثقًا من أن أحدًا غيره لن يكتشف ثغرة يوم الصفر، وإلا من الأفضل الكشف عنها.

بعض التحذيرات بشأن بياناتنا

في ما يلي شرح لبعض التحذيرات المتعلقة ببياناتنا.

يمكن تعميم نتائج بياناتنا على مجموعات بيانات مماثلة فحسب

تقوم استنتاجاتنا على مجموعة البيانات التي وصفناها في الفصل الأول. ونعتقد أن هذه البيانات تمثل نسبيًا ما تضمنه دولة قومية متقدمة في ترسانتها. وعليه، من المرجح أن يخلص قارئ يتعامل مع بيانات مماثلة إلى الاستنتاجات نفسها أو استنتاجات مشابهة حيال بياناته. ورغم ذلك، قد يكون تطبيق تعميم شامل على مجموعات البيانات الأخرى مضللًا، لأن التعميم على بيانات أخرى لا يصلح إلا إن كانت البيانات مطابقة من حيث طبيعتها ببياناتنا.[11]

قد يكون "الموت" مزيجًا من عدة أشياء

لا تعتبر برمجية إكسبلويت ثغرة واحدة فحسب بالضرورة. فقد تتضمن برمجية إكسبلويت ناجحة عدة ثغرات مترابطة، لذا لا يعني "قتل" برمجية إكسبلويت بالضرورة أنه جرى اكتشاف جميع الثغرات الموجودة في البرمجية. عوضًا عن ذلك، قد يعزى موت برمجية إكسبلويت إلى اكتشاف ثغرة أو مكون واحد. وعلى نحو مماثل، قد يكون إحياء برمجية إكسبلويت بسيطًا بقدر إيجاد بديل عن المكون المكتشف، ما قد يكون صعبًا أو بدون قيمة. ويركز هذا التقرير على برمجيات إكسبلويت، عوضًا عن الثغرات، ولا تعتمد حالة برمجية إكسبلويت بالضرورة على حالة الثغرة الكامنة.[12]

قد تكون بعض برمجيات إكسبلويت الحية ميتة في الواقع

قد تكون بعض برمجيات إكسبلويت الحية (والمجهولة) ميتة (معلومة للعامة) في الواقع. ففي حال وجد البائع المتضرر ثغرة، لا يحمل هذا الأمر معاني كبيرة بالضرورة. علاوةً على ذلك، قرر بعض الباحثين عدم الكشف عن الثغرات التي وجدوها.

[11] يصح ذلك على وجه الخصوص في البيئات التي تتبنى بسرعة أجهزة إنترنت الأشياء.

[12] مثلًا، قد نجد برمجية إكسبلويت تكون عبارة عن مزيج من الثغرات أو الأسس التي تتيح تحميل ملف أو قدرة الكتابة، كما هو الحال مع ثغرة تسمح بتنفيذ الرمز عن بعد. في حال اكتشاف ثغرة الكتابة الأساسية، تعتبر برمجية إكسبلويت ميتة حتى في حال عدم رصد الثغرة الأخرى.

هناك عدة أسباب تدفع بمطور برمجية إكسبلويت أو بصائد أخطاء اكتشف ثغرة إلى عدم الإفصاح عنها:

- يأتي العديد من الباحثين الأمنيين الذين يعملون لصالح منظمات تعنى بالمكافآت الأمنية أو بالأبحاث الأمنية من منظمات تركز على المنحى الهجومي أو التشغيلي، لذا فهم يدركون أهمية الاحتفاظ بثغرة.
- قد يجد الباحثون الأمنيون أو صائدو الأخطاء ثغرة مع شخص آخر (أو لصالح شخص آخر، مثلًا منظمة هجومية ضمن وكالة حكومية)، وفي حال كشفوها يكونون قد تخلوا عن قدرات باحث أو منظمة أخرى.
- قد يتعلق الأمر بشخصية الباحث الأمني. فيعتبر الإفصاح في بعض دوائر تطوير برمجيات إكسبلويت أمرًا غير لائق. وفي حال حدوث أي مشاركة، قد يحدث ذلك ضمن بعض الدوائر الخاصة، ربما من أجل مشاركة المعلومات أو إثبات انعدام الأمن أو لأغراض التباهي والافتخار.[13]
- قد يحتفظ الباحثون الأمنيون بالثغرة لاستعمالها في سياقات القرصنة التي تعود بجوائز نقدية كبيرة (رغم الخلاف بشأن ما إذا كان ذلك يحصل فعلًا).
- ينتظر الباحثون الأمنيون صيانة قاعدة الرموز للوصول إلى نهاية الحياة (مثلًا ويندوز أكس بي Windows XP) ربما قبل استعمال برمجية إكسبلويت ونشرها.
- قد يعتبر بعض الباحثين أن عملية الإفصاح لبائع معين مليئة بالتحديات، لذا لا بد من الإحجام عن ذلك.

انطلاقًا من هنا، في ما يخص بياناتنا، قررنا أن نكون محافظين عند تسمية الثغرات بأنها "حية". في حال ذكر أحد ما، عن قصد أو بدون قصد، الثغرة التي هي موضوع النقاش (حتى إن لم يشيروا إلى أنها ثغرة)، نعتبرها "ميتة".

تحذيرات أخرى حول البيانات

في حين تحدثنا إلى آخرين يملكون بيانات مشابهة ويتمتعون بخبرة مماثلة لجهة تطوير برمجيات إكسبلويت، ورغم أننا جمعنا البيانات من مصادر مختلفة، تلقينا بيانات من مصدر واحد فقط. وعليه، تنحصر بياناتنا في ثغرات من مصدر واحد، واعترفنا أنها قد لا تشمل جميع احتمالات الاستغلال.

يكون المشاركون في أسواق ثغرات البرمجيات في بعض الأحيان متخصصين. فقد يركز أحد ما على المتصفحات والبرمجيات القائمة على العميل، في حين قد يركز آخر على تطبيقات قائمة على الويب وبرمجيات من جانب الخادم. أما الآخرون فلا يعتمدون التمييز إلى هذا

[13] مثلًا، تعتمد برمجية إكسبلويت موجودة ضمن مجموعة بياناتنا جرى تصنيفها على أنها حية، على ثغرة ضمن فئة شبه مندثرة (مثلًا ثغرات الخط)، لكن لا يبدو أنه تم الإفصاح عن الثغرة (بما في ذلك من قبل باحثين متخصصين في هذه الأنواع المحددة من الثغرات ويكسبون رزقهم منها وباحثين اكتشفوا الثغرة لكنه يبقون المعلومات سرية).

الحد.[14] وبما أن بياناتنا تأتي من مصدر واحد، لا تمثل جميع الثغرات أو أنواع المنتجات (مثلًا رمز أصلي مقابل رمز تجاري جاهز، ومنتجات من جانب الخادم مقابل جانب العميل، وبرمجيات مقابل برامج ثابتة مقابل أجهزة، وأنظمة تشغيلية مقابل متصفحات).

ورغم وجود كمية ملفتة من مدخلات البيانات، قد يعتبرها البعض ضئيلة بسبب العدد القليل من البائعين والمنصات وأنواع الثغرات الممثلة (64 و12 و42 على التوالي)، وكان لا بدّ من إلغاء بعض برمجيات إكسبلويت والمعلومات بسبب الطبيعة الحساسة للعمل والعمليات الجارية. هذا وقد كانت قدرتنا على دراسة عدة خصائص تتمتع بها الثغرات في آن واحد محدودة، بما أن عدد حالات الوفاة الملحوظة أصبح قليلًا جدًا عند تصنيف الثغرات ضمن عدة فئات استنادًا إلى عدد كبير من الخصائص.

وقد تظهر بياناتنا أيضًا ترابطًا قويًا بين خصائص الثغرات.[15] على سبيل المثال، بما أن معظم هيكليات البيانات وتخصيصات الذاكرة في نواة لينكس قائمة على التكوين، تميل ثغرات فساد الذاكرة ضمن هذه النواة نحو التدفقات المفرطة المكونة. علاوةً على ذلك، نظرًا إلى أن لينكس مفتوحة المصدر ومايكروسوفت (مثلًا) مغلق المصدر، قد يتمتع عدد أكبر من الثغرات ضمن لينكس بوضع "عامل إعادة هيكلة الرمز"، في حين يتمتع عدد أكبر من ثغرات مايكروسوفت بوضع "خالدة".

قد تبرز أسباب تاريخية لموعد اكتشاف بعض الثغرات واستغلالها. على سبيل المثال، قد تتمتع الثغرات الطويلة الأمد بحياة طويلة لأن لا أحد يبحث عنها. كان الناس يبحثون قبل بضع سنوات، بشكل خاص وعام، عن هذه الثغرات لكن قد يكون تركيزهم قد تبدل مع مرور الوقت. وقد يكون هذا هو السبب خلف بقاء الثغرات الطويلة الأمد غير المكتشفة سرية، وعليه يعتقد البعض أنه يجب الانتظار طيلة مدة حياة البرمجية، إلى جانب التركيز على فئة كاملة من الثغرات واكتشافها وتخفيفها لمعرفة أي منها تتمتع بحياة طويلة.

أبحاث المتابعة

يسعى هذا البحث إلى أن تكون نقطة انطلاق للبحث عن ثغرات يوم الصفر وبرمجيات إكسبلويت من خلال البيانات وهي أداة ترمي إلى مساعدة أولئك المشاركين في المناقشات. لقد تطرقنا فقط إلى ظاهر الأمور لجهة ما يمكن التحقيق فيه من خلال البيانات، علمًا بأنه هناك المزيد من مجالات البحث. على سبيل المثال:

- ما هي خصائص (إن وجدت) ثغرات يوم الصفر التي تشير إلى حياة طويلة أو قصيرة، وما هي معدلات الاصطدام بين مجموعتين متشابهين؟ مع حصولنا على المزيد من البيانات، قد يصبح الغوص في هذا السؤال سخيفًا نسبيًا (بما أنه سبق أن وضعنا آليات التحليل)
- هل يمكن الحصول على مؤشر موثوقية لبرمجيات إكسبلويت بغية تحديد جودة البرمجية؟

[14] للمزيد من المعلومات عن مختلف أسواق ثغرات يوم الصفر، انظر الفصل الرابع الخاص بليبيكي وأبلون وويب، 2015.

[15] قمنا بدراسة بعض هذه الخصائص ضمن هذا التقرير، في حين ستجري دراسة أخرى خلال أبحاث مستقبلية.

- ما هو التداخل القائم بين دول قومية؟ ما هي أحجام وقدرات الدول القومية الأخرى؟
- كيف يمكن مقارنة بياناتنا مع مخزونات الثغرات الخاصة للشركات وقواعد البيانات الأخرى؟
- كيف يمكن مقارنة ما اكتشفناه مع ما اكتشفه خصومنا (أو أولئك الذين لديهم محفزات مماثلة)؟ هل يمكن لمجموعة ثغرات واحدة (مثلًا أفصحت عنها حكومة أو منظمة) المساعدة على معرفة الثغرات التي يحتفظ بها آخرون وكيف يؤثر ذلك على معدلات الديمومة أو الاصطدام؟
- ما هي الاتجاهات السائدة مع مرور الوقت؟ كما رأينا، فإن طبيعة الثغرات ديناميكية، وقد تؤثر عمليات التخفيف الجديدة على ديمومة برمجيات إكسبلويت وتحول التركيز نحو مطوري برمجيات إكسبلويت. إن أصبح الدفاع هو المسيطر فجأة، كيف يؤثر ذلك على الحاجة لاكتشاف الثغرات واستحداث برمجيات إكسبلويت لاستغلالها؟ هل يجب تغيير مخصصات الموارد؟
- ما معدل التداخل بين المكافآت الأمنية والذين يكتشفون الثغرات للاستغلال الخاص؟ يرى بعض الخبراء وجود تداخل كبير، في حين لا يفعل آخرون (مثلًا، أفاد باحث ثغرات تحدثنا إليه: "لم ألعن يومًا برنامج المكافأة الأمنية لإقدامه على تخريب خططي".) ما هو التأثير الطويل الأمد للمزيد من أبحاث الثغرات الاستقصائية واختبارات المكافآت الأمنية وحصدها؟
- ما هي فوارق التحليل بعد مرور فترة من الزمن (مثلًا عشر سنوات اعتبارًا من الآن) تحمل بيانات مماثلة، وهل يسلط ذلك بعض الضوء على ما إذا كانت برامج المكافأة الأمنية تخفض أمد الحياة المتوقع؟
- ما هي مستويات المسؤولية المناسبة التي يجدر بالحكومات أخذها في الحسبان؟ على سبيل المثال، في حال تكتمت الحكومة عن ثغرة استفاد منها المهاجمون للحصول على تسوية ناجحة من بنية تحتية أو شركة مهمة، هل تتحمل الحكومة بعض المسؤولية في هذا الشأن؟
- هل من طرق أفضل تتيح للباحثين الأمنيين والبائعين والحكومات مشاركة المعلومات بطريقة استباقية تسمح بإبقاء الثغرة حية من قبل الباحثين أو الحكومات، لكنها في الوقت نفسه تمكن الشركة المتضررة من حماية نفسها بشكل أفضل؟ هل من وسيلة لتوفير سلوك أو توقيع بدون كشف كافة المعلومات المتاحة عن ثغرة يوم صفر؟

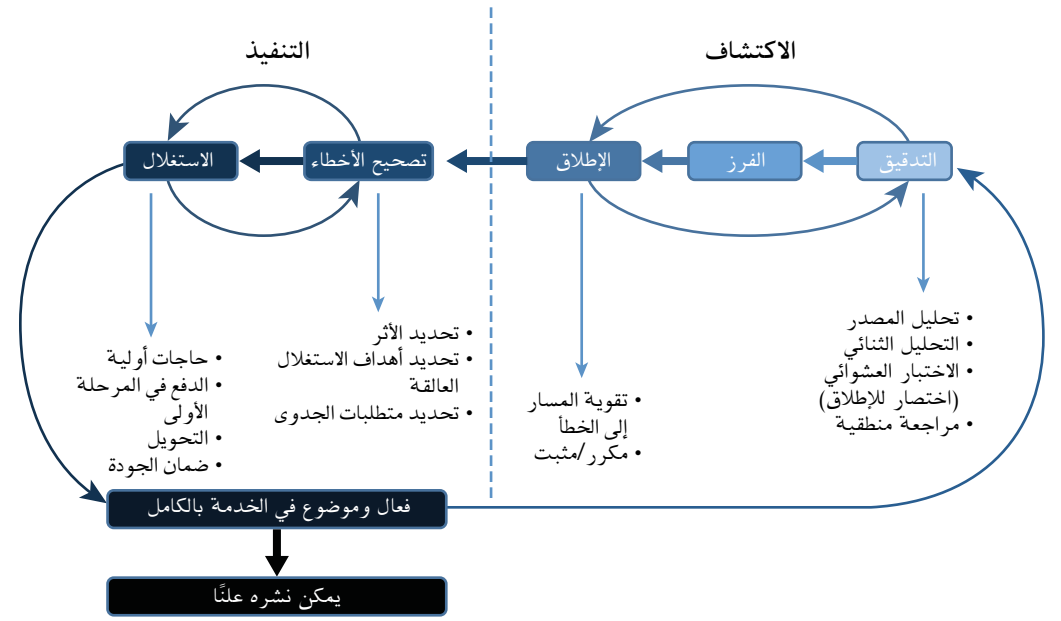
دورة تطوير برمجية إكسبلويت

تتألف عملية تطوير برمجية إكسبلويت من عدة خطوات، وقد تسلك كل خطوة عدة مسارات.

المرحلة 1: الاكتشاف والتحقق

في المرحلة الأولى، يبحث الباحثون عن الثغرات ويطورون برمجية إكسبلويت موثوقة قائمة على إثبات المفهوم.

الشكل A.1
دورة حياة تطوير برمجية إكسبلويت



RAND *RR1751-A.1*

الخطوة 1: التدقيق واستحداث حوادث الاصطدام

تتمثل الخطوة الأولى لاستحداث برمجية إكسبلويت في اكتشاف ثغرة يجدي نفعًا تطوير برمجية إكسبلويت لها (نطلق عليها تسمية ثغرة "جديرة بالاستغلال"). يتم إطلاق خطأ أو مشكلة خارجية، ما يسبب حادث اصطدام بالنسبة إلى الباحث عن الثغرات يحثه على مواصلة التحقيق.

إنها عملية تدقيق بالرمز، قد تكون سطحية نوعًا ما (مثلًا تشغيل "اختبار عشوائي" تجاري على منتج)،[1] أو عميقة إلى حدّ كبير (مثلًا تحليل يدوي للرموز الثابتة والديناميكية أو تحليل العيوب).[2] وتتباهى بعض المجموعات التي تعنى بأبحاث الثغرات بعدم استعمالها الأدوات الآلية، حيث تزعم أن استراتيجية التدقيق اليدوي تعتبر إحدى ميزاتها لاكتشاف الثغرات المستترة.

ويمكن أن يكون لأسلوب اكتشاف الثغرات أثر على أي ثغرات تم اكتشافها في الواقع. وعلى سبيل المثال، زعم بحث جرى مؤخرًا أن الاختبارات العشوائية لا تكتشف سوى الجزء الظاهر من الثغرات (كيردا Kirda وآخرون، غير مؤرخ). ويمكن اكتشاف الثغرات من خلال الاختبار العشوائي لمنتجات أحدث أو أقل نضوجًا، أو تلك التي لديها قواعد رموز بسيطة. وبالنسبة إلى المنتجات التي تتمتع بحياة أطول، أو التي هي أكثر تعقيدًا، أو التي تعرف بحصتها السوقية الكبيرة، أو التي تدر إيرادات مرتفعة، فقد قيّم المزيد من الأشخاص قواعد الرموز، علمًا بأن اكتشاف الثغرات غالبًا ما يتطلب تدقيقًا أعمق ومراجعة منطقية إضافةً إلى تحليل رمز المصدر بغية الغوص في عدة طبقات.

ويختلف احتمال اكتشاف ثغرة جديرة باستحداث برمجية إكسبلويت لها بحسب عدد الأشخاص الذين سبق لهم أن درسوا هذا الرمز وعمق تحليل الرمز (إذ يتم اكتشاف الثغرات السطحية أولًا) وتعقيد الرمز[3] ومعدل تغيير المنتج[4] ونضوج المنتج[5] ووظيفة المنتجات.[6]

كما أصبح النفاذ إلى اصطياد الأخطاء أسهل حيث إن الموارد (كتب وكتيبات عبر الإنترنت ومحادثات) باتت متاحة لأي طرف يهمه الأمر.

ويركز بعض الباحثين عن الثغرات على بائعين محددين في عملية التدقيق في الرموز،[7] في حين يبقى آخرون منفتحين على التدقيق في مجموعة من المنتجات. أما البعض الآخر،

---

[1] لا يجري استحداث جميع الاختبارات العشوائية بالتساوي. فبعضها قدر على الغوص بشكل أعمق من سواها، والبعض الآخر يجد أخطاء مختلفة بحسب الأسلوب المتبع. على سبيل المثال، يكتشف اختبار American Fuzzy Lop الأخطاء بسرعة من خلال تعقب كل "بيان افتراض" في قاعدة رموز.

[2] يتتبع تحليل العيوب بيانات المراقبة التي تمّ إدخالها ضمن تدفقات الرموز لتعقب حركتها في عملية ما. انظر نيوسوم (Newsome) وسونغ (Song) (2005) للمزيد من الأبحاث القائمة على تحليل العيوب.

[3] كلما ازداد تعقيد الرمز، ازدادت فرص بروز مشاكل المواءمة والثغرات القابلة للاستغلال.

[4] مثلًا، يعمد بائعو المتصفحات باستمرار إلى تحديث الرمز وإطلاق رموز جديدة، وبالتالي إدخال أخطاء جديدة بوتيرة أسرع. في المقابل، على صعيد منتجات الشبكة، على غرار الموجهات، هناك جدول زمني أطول لإطلاق نسخ جديدة، كما يستغرق إصدار الميزات وقتًا أطول ويكون التدفق بطيئًا، وعليه، يجري إدخال الأخطاء الجديدة بوتيرة أبطأ.

[5] كلما كان المنتج أقدم، كلما ازداد احتمال اكتشاف الأخطاء السطحية. علاوةً على ذلك، إن كان منتج يتمتع بتاريخ من الرموز الشائبة، فمن المرجح أن يحتوي على رموز مماثلة في المستقبل.

[6] غالبًا ما يكون إجراء الاختبار العشوائي على الأخطاء أسرع في المنتجات التي تقوم بوظيفة واحدة وتعيد الكرة، كأجهزة قراءة الصور وأجهزة تحليل الفيديوهات والموسيقى، إلخ

[7] إنها بشكل خاص حالة الشركات التي تضم مجموعات داخلية تعنى بالبحث عن الثغرات، حيث إن البحث في منتجاتها الخاصة منطقي. في هذا الإطار، ذكرت إحدى الشركات أنها تركز على الأهداف المهمة مثل المنتجات في الأسواق الناشئة أو المنتجات التي تتمتع بحصة سوقية كبيرة.

فبدلًا من التركيز على بائع أو منتج محدد، يبحثون عن نمط ثغرات معين بغض النظر عن المنتج الموجودة فيه.[8]

الخطوة 2: فرز حوادث الاصطدام لتحديد كيفية حدوثها

شملت عملية الفرز تعقب حادث اصطدام لتحديد السبب الجذري في الرمز، أي معرفة أي جزء في الرمز تسبب بالحادث. أحيانًا، يكون هذا الجزء من الرمز مفيدًا للتحكم بأكثر من حادث اصطدام. وتسمى هذه الأجزاء عناصر أساسية لأنها بمثابة وحدات بناء الثغرات ويمكن غالبًا استخدامها في عدة برمجيات إكسبلويت.[9]

الخطوة 3: تطوير عملية إطلاق موثوقة

فور اكتشاف كيفية وقوع حادث الاصطدام، على الباحث تحديد كيفية تقوية المسار إلى الثغرة، وضمان إمكانية إطلاق الحادث بشكل موثوق عند الحاجة. في هذه المرحلة، يجري استحداث برمجية إكسبلويت قائمة على إثبات المفهوم.

في أغلب الأحيان، يتوقف عمل صائدي الأخطاء الذين يستهدفون اكتشاف الثغرات ومشاركة المعلومات أو بيعها للبائع المتضرر أو لبرنامج مكافأة أمنية في هذه المرحلة.

المرحلة 2: التطبيق

خلال المرحلة الثانية، يستخدم الباحثون عن الثغرات برمجية إكسبلويت القائمة على إثبات المفهوم وينتجون برمجية إكسبلويت فعالة بالكامل (إن أمكن).

الخطوة 4: تصحيح الأخطاء وتحديد متطلبات الاستغلال الكامل

في هذه المرحلة، يقيّم الباحث عن الثغرات مختلف حوادث الاصطدام القابلة للإطلاق والموثوقة من أجل تحديد أي منها قابل للتطوير ليصبح برمجية إكسبلويت فعالة بالكامل (بدلًا من أن تكون مجرد إثبات للمفهوم). فور حصر الخيارات، يحدد الباحث ما الذي يمكن فعله تمامًا بشأن حادث الاصطدام (مثلًا تنفيذ الرمز عن بعد أو تسريب المعلومات أو سحق عملية مختلفة) والأثر المحتمل الناتج عن حادث الاصطدام والمتطلبات الأخرى الضرورية لتقليص مدى الشكوك واستحداث برمجية إكسبلويت فعالة بالكامل (مثلًا وحدات بناء أخرى). في حال اعتبرت متطلبات أخرى ضرورية، قد يحتاج الباحث إلى العودة إلى المرحلة الأولى من أجل اكتشاف المزيد من الثغرات والتخلص من الغموض.

[8] يمكن تحقيق مكاسب كبيرة عبر هذه الاستراتيجية التي تركز على أنماط محددة، لكنها تتضمن مخاطر عالية في الوقت نفسه. فتأتي المكاسب الكبيرة من أن تطوير برمجية إكسبلويت يكون سطحيًا فور الاكتشاف بينما يأتي الخطر الأكبر من أنه في حال اكتشف شخص آخر النمط، يمكن إصلاحه وإلغاء قدرة الباحث بضربة واحدة. وقد أشار باحث عن الثغرات إلى أن هذه الأنماط شكلت أحد الجزئين المهمين من حقوق الملكية الفكرية التي يتمتع بها.

[9] تشمل أمثلة عن العناصر الأساسية أجزاء من الرمز تسمح بتصعيد الميزة أو كتابة القدرة أو قراءتها أو تنفيذها.

في أغلب الأحيان، قد تتألف برمجية إكسبلويت من عدة ثغرات تساهم كل منها في الاستغلال الكامل. على سبيل المثال، قد تتطلب برمجية إكسبلويت وجود ثغرة حالة تسابق (race condition vulnerability) بغية الاستفادة عندها من ثغرة سوء إدارة الذاكرة. ومن الواضح أنه كلما كان عدد الثغرات الضرورية أقل، كان استحداث برمجية إكسبلويت فعالة بالكامل أسهل، وكلما ازداد عدد المتغيرات ازدادت صعوبة الاستغلال.

الخطوة 5: استحداث برمجية إكسبلويت كاملة

ما إن يحدد الباحث عن الثغرات متطلبات الاستغلال الكامل (أي يضع نظرية الاستغلال) ويكون قد حصل على وحدات البناء الضرورية، يبدأ أو تبدأ بتطبيقها لإزالة أي غموض قائم. في بداية الأمر، قد يركز مطور برمجيات إكسبلويت على الاحتفاظ بتنفيذ الرمز أو موطئ قدم أولي في العملية (من خلال ضخ غرض ما أو فساد الذاكرة). في نهاية المطاف، يقوم/تقوم الباحث/الباحثة بإضافة قدرات للاستمرار أو للتحويل إلى بنية تحتية ومنصات مختلفة،[10] أو ضمان المواءمة مع نسخ أخرى،[11] أو إزالة أي مؤشر على وجود برمجية إكسبلويت في النظام.[12] تبرز أحيانًا الحاجة إلى ربط برمجيات إكسبلويت ببعضها أو إلى تلبية ظروف تشغيلية محددة لإحداث الأثر المرجو.[13] والآن يكون الباحث عن الثغرات قد حصل على سبيل برمجية إكسبلويت فعالة بالكامل.

المرحلة 3: النقل التشغيلي

الخطوة 6: الاستخدام في العلن

لاستعمال برمجية إكسبلويت من الناحية التشغيلية، غالبًا ما يجري اتخاذ المزيد من الخطوات. فاستعمال برمجية إكسبلويت "في العلن" قد يختلف عن استعمالها ضمن إطار المختبر. ففي الإطار الأخير، تبقى البيئة الاختبارية نفسها، حيث تتم مراقبة حمولة الشبكة ومعدل التباطؤ مع الحرص على اتساقها، كما يستخدم النظام المستغل لأغراض الاختبار فقط. في بعض الأحيان، قد تؤيد جهود إزالة الأخطاء في المختبر إلى إدخال سلوك خاص بالعملية ما كان ليكون قائمًا في بيئة تشغيلية. أما ضمن الإطار التشغيلي، فقد يكون النظام المستهدف خاضعًا لحمل ثقيل، حيث يجري تحميل مختلف حزم اللغات ومكتبات الربط الديناميكية بشكل مختلف. وبالتالي، قد تكون أي فرضيات قائمة على المختبر وتتعلق بتصاميم ذاكرة العملية غير صالحة أبدًا ضمن السياق التشغيلي. ومن شأن ذلك أن يؤثر بشكل كبير على طريقة عمل برمجية إكسبلويت.[14]

[10] ما يتطلب أحيانًا العودة إلى عملية التدقيق الأولية.

[11] مثلًا في خادم ويندوز 2000، يمكن إجراء الاستغلال عن بعد، لكن في خادم ويندوز 2003، يتم إجراؤه محليًا بشكل عام.

[12] مثلًا، تنظيف الذاكرة المستخدمة في حال إجراء الاستغلال ضمن نطاق النواة.

[13] مثلًا، قد يتطلب استغلال ثغرة في حالة سباق التواجد في نظام مستهدف مثير للاهتمام لعدة دقائق بهدف المواءمة مع عدة عمليات.

[14] مثلًا، بالنسبة إلى برمجيات إكسبلويت لاستغلال فساد الذاكرة، قد يعتمد الاستغلال الناجح على معرفة كيفية استخدام التكوين وكيفية معالجة طلبات النظام الأخرى بالتزامن مع طلبات نظام برمجيات إكسبلويت.

نتيجة لذلك، غالبًا ما قد يعتمد نجاح برمجية إكسبلويت في سياق تشغيلي على خبرة المشغل وتدريبه.[15]

ويعتبر نشر الأداة (أو المغروس أو حمولة مرحلة n) الخطوة التالية للاستغلال، غير أنها ليست جزءًا من عملية التطوير.

مواكبة العصر

يعتبر تطوير برمجية إكسبلويت عملية مكررة. فالكثير من مطوري برمجيات إكسبلويت، الفخورين بعملهم، سيواصلون عمليات التدقيق لضمان فعالية برمجيات إكسبلويت الخاصة بهم عند إطلاق نسخ جديدة للبرمجيات رغم أن العملاء لا يطلبون ذلك دائمًا. وقد تتفاقم هذه المشكلة مع مرور الوقت.

فغالبًا ما يواجهون برمجية إكسبلويت جرى اكتشافها (أو تسريبها) في العلن، وعليهم مراقبة ما إذا كانت أي برمجية إكسبلويت خاص بهم قد شاركت أي من العناصر الأساسية المستخدمة من قبل البرمجية.

تُستخدم نسبة ضئيلة فقط من حوادث الاصطدام أو الثغرات المطلقة باعتبارها ثغرات قابلة للاستغلال.

رغم تنامي شعبية برامج المكافأة الأمنية وازدياد عدد صائدي الأخطاء، قد يكون اكتشاف ثغرات قابلة للاستغلال بين الأخطاء مليئًا بالتحديات، لأن مجموعة صغيرة من الأخطاء فقط قابلة في الواقع للاستغلال بشكل مفيد. ما من نسبة شاملة للثغرات القابلة للاستغلال من الأخطاء، فالأمر يتوقف على البائع ونوع المنتج (مثلًا المتصفحات مقابل الموجهات) وممارسات تطوير البرمجيات، إلخ. وقد يقول البعض إنه يجب اعتبار أي حادث اصطدام نوعًا من الهجوم وبالتالي قد يكون مفيدًا في إطار تشغيلي، لكن من المعروف عادة أن نسبة ضئيلة فقط من الثغرات المكتشفة من خلال التدقيق (الاختبار العشوائي اليدوي أو الآلي) هي في الواقع "مفيدة" للاستغلال. وقد منحنا الأشخاص الذين تحدثنا إليهم مجموعة تقديرات لنسبة حوادث الاصطدام أو الأخطاء المطلقة التي قد تصبح ثغرات مفيدة وقابلة للاستغلال: 1%-0.5 و20%-15 و25- 30%. وقد أبلغنا أحد الباحثين عن ثغرات أن نسبة كبيرة، وصلت إلى 50% من الثغرات التي اكتشفها، يمكن استغلالها، مشيرًا إلى أن الأسلوب الذي اعتمده لاكتشاف الثغرات خفض عدد الثغرات غير المجدية، وأنه حاول استغلال كل ثغرة اكتشفها.

يجري أحيانًا الاحتفاظ بالثغرات التي تصنف أنها غير قابلة للاستغلال "للحالات الطارئة". ففي حال بدأ باحث جديد عن الثغرات العمل في شركة، أو إن كان أحد يبحث عن وظيفة ما، قد يعمد إلى مراجعة الثغرات لمراقبة أي تغييرات في الرمز قد تجعلها قابلة للاستغلال.

[15] مثلًا، الإقرار بأنه قد يكون من الضروري نشر برمجية إكسبلويت في نظام مستهدف بعد ساعات، حين تكون حمولة الشبكة ومعدل التباطؤ عند أدنى مستوى.

بعض العوامل التي قد تؤثر على تطوير برمجيات إكسبلويت

قد يكون لبعض عمليات التخفيف أثر كبير

تؤثر عمليات التخفيف على قابلية الاستغلال، وقد تخلف أثرًا كبيرًا على ما يجري استغلاله. وبحسب التوقعات، تلعب عمليات التخفيف الأمنية والتدابير المضادة دورًا كبيرًا في استغلال ثغرة ما. وبالتالي، قد يتغير ما هو قابل للاستغلال وما يتمّ استغلاله مع مرور الوقت.

على سبيل المثال، كان يعتبر استغلال الثغرات الموجودة في النواة في الماضي أسهل من استغلال تلك الموجودة في مجال المستخدم، ويعود السبب عمومًا إلى تقليص التعقيد في النواة (مثلًا الإسناد المؤشري الباطل).[16] وما إن بدأت الشركات تولي أهمية أكبر إلى أمن النظام التشغيلي، تم إدخال عمليات تخفيف جديدة إليها، ما صعب استغلال النواة.[17]

قد يكون التاريخ – جودة الرمز وتاريخ قاعدة الرمز – مؤشرًا جيدًا على قابلية الاستغلال

يعتبر التاريخ مؤشرًا جيدًا على قابلية الاستغلال. ففي حال اكتشاف العديد من الثغرات وبرمجيات إكسبلويت في الماضي ضمن قاعدة رموز محددة، ثمة احتمال كبير أن يتم اكتشاف العديد من الثغرات ضمن عمليات مكررة مستقبلية في قاعدة الرموز هذه. وقد يعزى ذلك إلى أن (1) قاعدة الرموز مليئة بالثغرات، و/أو (2) قاعدة الرموز تتمتع بحصة سوقية كبيرة ويعمل عدد أكبر من الباحثين على مراقبتها واكتشاف الثغرات فيها. وفي بعض الأحيان يحذر الباحثون نظرائهم في مجتمع أمن المعلومات بشأن قواعد الرموز التي تتضمن ثغرات.[18]

الأدوات والمعلومات الخاصة بتطوير برمجيات إكسبلويت متوافرة إلى حدّ كبير، ما يتيح القرصنة الجماعية

أصبحت الموارد متاحة بشكل أكبر للمساعدة على إضفاء طابع رسمي على البحث عن الثغرات وعمليات تطوير برمجيات إكسبلويت. بين عامي 2007 و2010، باتت الكتب والمحادثات والفئات والموارد الإلكترونية متاحة على نطاق واسع لتعليم أي شخص كيفية اكتشاف الثغرات واستغلالها.[19] انطلاقًا من هنا، بدأ العديد من الأشخاص الذين يفتقرون إلى خلفية رسمية بدخول عالم تطوير برمجيات إكسبلويت.

---

[16] مثلًا، ضمن مجال المستخدم، على المرء العمل مع المخصصات الديناميكية ومختلف النسخ المجمعة، إلخ في حين أنه في النواة، ثمة توزيع مجمع واحد.

[17] تشير بعض التقديرات إلى أن التحول في مايكروسوفت قد حدث بين عامي 2007 و2008، وعلى مقربة من عام 2010 (عند اعتماد GRSecurity) بالنسبة إلى لينكس.

[18] مثلًا، خلال عرضه في مؤتمر ZeroNights 2013 E.0x03، حذر الباحث عن الثغرات ماتيوس جورشيك (Mateusz "j00ru" Jurczyk) الحاضرين من أن الثغرات الأمنية في النواة في ويندوز كانت وستبقى كثيرة (جورشيك Jurczyk، 2013، الشريحة 107).

[19] توفر شركة نو ستارش برس (No Starch Press) العديد من الكتب حول القرصنة وأمن الكمبيوتر ومواضيع البحث عن الثغرات (نو ستارش برس No Starch Press، غير مؤرخ). كما تقدم المدونات معلومات متعمقة حول تقنيات جديدة لاستغلال حالات السباق (مثلًا ستالك أر StalkR، 2010). ويقدم ألبرتس (Alberts) وأولداني (Oldani) (غير مؤرخ) مثلًا عن محادثات خلال مؤتمر تناولت استغلال نظام تشغيل أندرويد (Android). وعليه، كانت بعض الموارد متوافرة حتمًا قبل العام 2007. على سبيل المثال، تتضمن مدونة أندريز براور (Andries Brouwer) بعنوان "Hackers Hut" (براور Brouwer، 2013) معلومات وأمثلة عن تقنيات استغلال متنوعة (مثلًا حالات السباق والتكديس والتكوين).

يمكن لأي شخص أن يتعلم الطهي، لكن لا يمكن لأي كان أن يصبح رئيس طهاة

لا يعني توافر الموارد أنه بإمكان أي شخص أن يعدّ برمجيات إكسبلويت فعالة بالكامل. ويمكن للمحادثات الدائرة خلال المؤتمرات والتقارير والكتب أن تحدد كيفية استحداث برمجية إكسبلويت. لكن الاستفادة من هذه المعلومات وإعداد برمجية إكسبلويت لاستغلال ثغرة جديدة لا يزال يتطلب مهارات. وقد شبّه بعضهم هذا الأمر بعالم المطبخ، فيمكن لأي امرئ الحصول على الوصفة (برمجية إكسبلويت قائمة) وإعادة تصنيع هذا الطبق خطوة بخطوة، لكن يبقى فهم كيفية استعمال مكونات جديدة (ثغرات) والنكهات (تقنيات) لإعداد طبق جديد وشهي بالقدر نفسه (برمجية إكسبلويت جديدة) ميزة لا يتمتع بها الجميع.

فضلًا عن ذلك، ناهيك عن اكتشاف الثغرات وتطوير برمجيات إكسبلويت، يحتاج الباحث الماهر أيضًا إلى معرفة كيفية صياغة تقارير جيدة وإظهار إمكانيات برمجية إكسبلويت لعميل محتمل.

العنصر البشري: شخصية وقدرات المحلل

يعتبر العنصر البشري عاملًا أساسيًا في تطوير برمجيات إكسبلويت. فالشخصية والقدرة الفردية للمحلل الذي يدقق بالرمز والمطور الذي يعد رمز برمجية إكسبلويت والمشغل الذي يوظف منتج برمجية إكسبلويت تؤثران على النجاح الإجمالي للمنتج النهائي.

ويضفي كل باحث عن الثغرات مهاراته الخاصة، فيصبح الكثيرون ماهرين في منصة أو قاعدة رموز معينة أو في استعمال تقنية محددة (مثلًا آلية تحديد الوصول في ويندوز مقابل لينكس). وبالتالي، يمكن للتغييرات ضمن فريق تطوير برمجيات إكسبلويت أن تؤثر بشكل كبير على النتائج المتوقعة.

غير أنه لا يمكن اكتشاف أو استغلال جميع الثغرات على نحو متساوٍ، فقدرة الباحث عن الثغرات وشخصيته أمران مهمان. على سبيل المثال، يتطلب اكتشاف ثغرات حالات السباق واستغلالها القدرة على تصور عدة مسارات رموز متزامنة وتعقبها.

وقد تعتمد برمجيات إكسبلويت ذات الأمد الطويل (أي تلك التي تبقى ثغراتها مجهولة للعامة وحية لفترة طويلة) على قدرة مطور برمجيات إكسبلويت على الغوص في عدة طبقات ضمن مجالات الرمز أكثر منه خلال الاختبار العشوائي الطبيعي (الذي يجري عمومًا على السطح).

ويعتبر بعض الباحثين عن الثغرات أفضل من غيرهم في اكتشاف الثغرات القابلة للاستغلال، في حين يعتبر آخرون أفضل من حيث تنفيذ الاستغلال. وبالتالي، ينقسم الباحثون عن الثغرات في العديد من متاجر تطوير برمجيات إكسبلويت إلى ثنائيات أو يعملون ضمن فرق.

أخيرًا، بإمكان الإدارة الجيدة أن تساعد إلى حدّ كبير فريق البحث عن الثغرات ليحقق النجاح. ويعتقد البعض أنه يتعين على الإدارة الأفضل التمتع بخبرة سابقة في تطوير برمجيات إكسبلويت من أجل إيجاد أفضل الباحثين وتشكيل فريق جيد.

الملحق B

الباحثون عن الثغرات: من الذي يبحث عن الثغرات؟

لا يمكن اكتشاف الثغرات (أو استغلالها) بدون مشاركة بشرية.[1] وبالتالي، سعينا إلى فهم ما يمكن حيال اكتشاف الأشخاص للثغرات واستغلالها لمعرفة دور العنصر البشري في تطوير برمجيات إكسبلويت.

طول السير المهنية لصائدي الأخطاء والباحثين عن الثغرات وخبراتهم

إلى جانب المعلومات عن برمجيات إكسبلويت والثغرات التي تمّ استغلالها، شملت مجموعة بياناتنا معلومات عن 23 باحثًا عن الثغرات طوروا هذه البرمجيات. وقد ركز بعض الباحثين على منصة معينة (مثلًا، منصات محددة مفتوحة المصدر). في حين كان باحثون آخرون ماهرين في العمل على نوع معين من الثغرات (مثلًا حالات السباق ضمن لينكس). أما قسم آخر فلم يحصر علمه بنوع واحد، حيث تمكن من الانتقال بين قواعد الرموز وأنواع الثغرات. وخلال محادثاتنا مع متاجر أخرى لتطوير برمجيات إكسبلويت، علمنا أن بعض الباحثين يفضلون العمل ضمن ثنائيات، حيث يتولى أحدهما اكتشاف جميع الثغرات، بينما يفضل الآخر إتمام استغلال الثغرات.

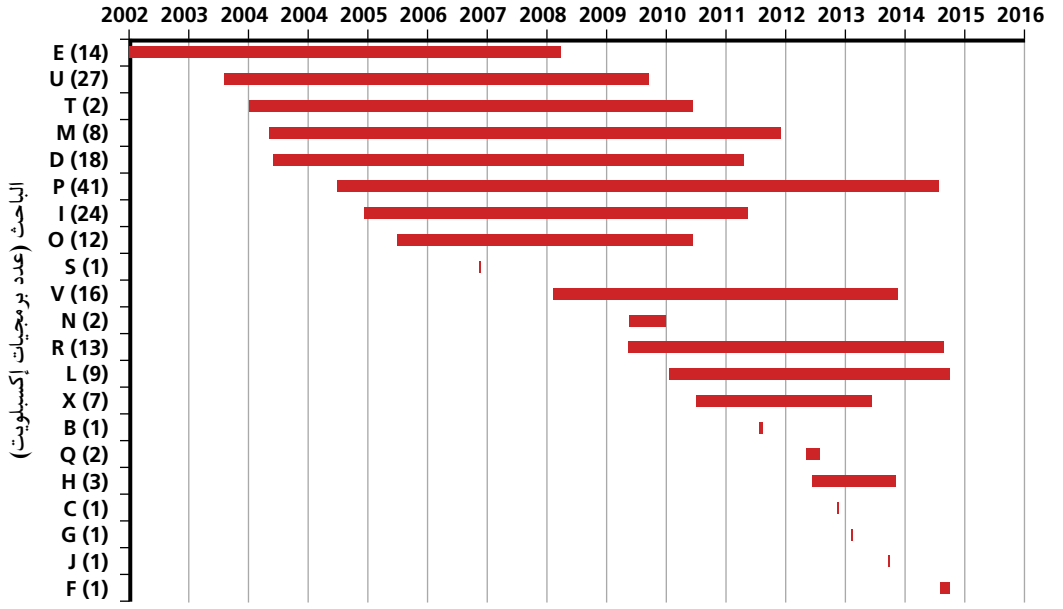
أمد السيرة المهنية للباحثين عن ثغرات

يعتبر البحث عن الثغرات عملًا قد يستنزف قوى الكثيرين (ليبيكي وأبلون وويب، 2015). فقد تفيد معرفة طول السيرة المهنية القابلة للنمو في أغراض التخطيط أو التوظيف. قمنا بدراسة خصائص الباحثين عن الثغرات ومقارنتها مع خصائص الباحثين الذين اكتشفوا ثغرات مفصح عنها علنًا، لا سيما صائدي الأخطاء في مايكروسوفت.

في الشكل B.1، قمنا بتحديد معالم طول السيرة المهنية لمطوري برمجيات إكسبلويت. ويتمثل تاريخ الانطلاق في التاريخ الأول الذي سجل فيه الباحث اكتشافه أو استغلاله لثغرة ما (أي تاريخ الميلاد أو النضوج)، في حين يكون تاريخ الانتهاء هو التاريخ الأخير الذي سجل فيه الباحث اكتشافه أو استغلاله لثغرة ما.

[1] يأمل التحدي الافتراضي الكبير (Cyber Grand Challenge) وجهود المتابعة من وكالة المشاريع البحثية الدفاعية المتقدمة (DARPA) (وكالة المشاريع البحثية الدفاعية المتقدمة، 2016) في تعطيل هذا المعيار، رغم أنه لا يزال يطلب من البشر تشفير الحلول.

الشكل B.1
طول السير المهنية للباحثين عن الثغرات ضمن BUSBY (n =21)



RAND *RR1751-B.1*

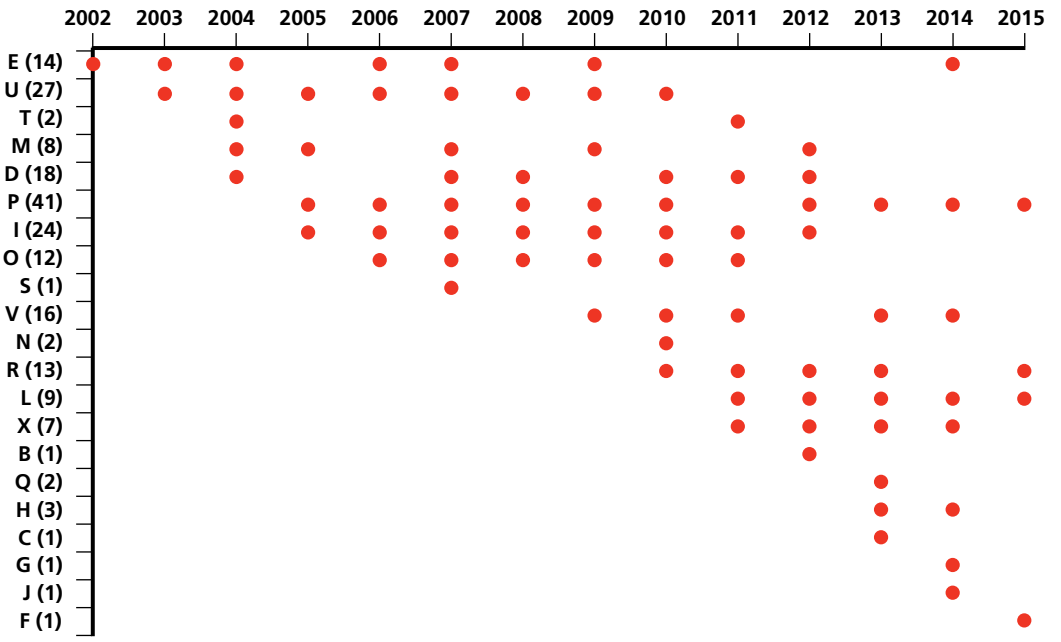
يظهر الشكل B.2 أن معدل التطوير ليس مطردًا، ففي بعض السنوات، قد لا يكتشف الباحث برمجية إكسبلويت أو يتمكن من تطويرها.

استنادًا إلى هذه البيانات، يبدو أنه ثمة تواتر عالٍ للمطورين في المراحل الأولى (مثلًا، الباحثين U وP وI) مقابل تواتر منخفض للمطورين في المراحل المتقدمة.[2] ويمكن التقدير أنه خلال السنوات الأولى، برزت العديد من الأهداف السهلة، أو أن مجموعة الباحثين بلغت ذروتها، ومن ثم خسرت بعض العناصر التي قدمت الأداء الأفضل، وهي الآن في طور التضاؤل بحيث باتت تضم مطورين أقل كفاءة. وعلى وجه الخصوص، قد يتمتع الباحثون U وP وI بلمسة خاصة بهم.[3]

[2] بسبب طبيعة البيانات (مثلًا جرى إلغاء بعض البيانات بسبب الحساسية التشغيلية) تكون هذه الأرقام غير مكتملة، وقد يتبين أن بعض الباحثين عن الثغرات (وبالتالي الفريق بكامله) يقدمون أداء أدنى مما تتطلبه الحالة فعليًا.

[3] قد يكون الباحث U استفاد من الكثير من "الأهداف السهلة". فقد طور برمجية إكسبلويت الأخيرة بحلول 2010، أي في وقت اتفق فيه الكثيرون ضمن مجتمع الباحثين عن الثغرات على أن عمليات تطوير برمجيات إكسبلويت والتخفيف الخاصة بالثغرات بدأت تتزايد بشكل كبير، وأصبح عدد أكبر من المعلومات بشأن البحث عن الثغرات وإيجاد الأخطاء متوافر للعامة، ما أدى إلى اكتشاف العديد من الثغرات بسرعة من قبل آخرين. أما برمجيات إكسبلويت التابعة للباحث P فقد تركزت جميعها على لينكس. ونتيجة ذلك، كان الباحث يتمتع بدراية كبيرة بالعناصر الداخلية ونواة لينكس، ما سمح له بإيجاد الثغرات المستترة إلى حدّ كبير. يذكر أن عدد برمجيات إكسبلويت الميتة تتجاوز بمرتين تلك الحية بالنسبة إلى الباحث U. غير أن ثمة شبه تعادل بين برمجيات إكسبلويت التابعة للباحث P التي ما زالت حية، وتلك الميتة.

الشكل B.2
السنوات التي اكتشف خلالها الباحثون الثغرات ضمن BUSBY (n =21)



RAND *RR1751-B-2*

على ما يبدو، أشار فريق آخر يعنى بالبحث عن الثغرات إلى أنه بالنسبة إليهم، ينقل كل باحث حوالى أربع برمجيات إكسبلويت عاملة بالكامل سنويًا. فقد يكتشف مطور المزيد من الثغرات وينتج المزيد من برمجيات إكسبلويت القائمة على إثبات المفهوم، لكن العديد من الثغرات المكتشفة قد لا تكون مفيدة أو لا تلبي متطلبات العميل.[4]

أمد حياة الباحثين عن الثغرات في الأسواق الرمادية أو الحكومية

أشارت تقديرات سابقة إلى دورة من ثلاث سنوات للباحثين عن الثغرات يكونون خلالها ذوي صلة من الناحية التقنية قبل البدء باكتساب المهارات، ما لم يعملوا بنشاط للبقاء متفوقين على صعيد إحراز تقدم جديد في ما يخص الرمز وعمليات التخفيف (ليبيكي وأبلون وويب، 2015). وفي حين يعتبر اصطياد الأخطاء وتطوير برمجيات إكسبلويت مهارتين مختلفتين، نقارن طول السيرة المهنية الخاصة بالباحثين عن الثغرات في الأسواق الرمادية أو الحكومية مع أمد حياة صائدي الأخطاء في الأسواق البيضاء (أي المكافأة الأمنية والبائعين ومجموعات على غرار المشروع صفر في غوغل).

[4] ما يثير الاهتمام هو أن صائدي الأخطاء من أصحاب القبعات البيضاء الذين يستثمرون وقتًا طويلًا في اكتشاف الثغرات يجدون على ما يبدو العدد نفسه بدورهم. وقد أعلنت باغ كراود في 2016، أن أبرز "صائديها الأقوياء"، أي الذين يكتشفون العدد الأكبر من الثغرات ويتلقون الأجر الأعلى، قدموا ما معدله 3.75 خطأ سنويًا (باغ كراود، 2016).

باستخدام بيانات النشرات الأمنية من مايكروسوفت بديلاً منطقيًا لصائدي الأخطاء في السوق الأبيض، نقوم بحساب متوسط أمد السيرة المهنية لصائد أخطاء.[5] ونقوم بالمثل بالنسبة إلى البيانات المتوافرة لدينا حول مطوري برمجيات إكسبلويت، كما نعمد إلى مقارنتها. وفي الحالتين، اكتشف بعض الصائدين والباحثين (وسجلوا) ثغرة واحدة. هذا وقارنا أمد الحياة مع أولئك الذين اكتشفوا ثغرة واحدة وبدونهم. ويظهر الجدول B.1 المقارنة. فصائدو الأخطاء الذين اكتشفوا أكثر من ثغرة تمتعوا بسيرة مهنية ناهزت العامين والنصف، في حين كانت سيرة مطوري برمجيات إكسبلويت المهنية أطول بشكل كبير أي دون أربع سنوات ونصف بقليل. وقد يعزى السبب إلى أن صائدي الأخطاء يتنقلون بين المنتجات (لذا لا يكفي النظر إلى بيانات النشرة الأمنية لمايكروسوفت لتقييم طول سيرتهم المهنية)، أو إلى أنهم يشغلون وظائف أخرى في مجال أمن المعلومات (أو غيره) واصطياد الأخطاء هو مجرد هواية بالنسبة إليهم.

الجدول B.1
متوسط طول السيرة المهنية المسجلة لصائدي الأخطاء ومطوري برمجيات إكسبلويت في BUSBY

مع الباحثين عن "ثغرة واحدة أو بدونهم"	صائدو الأخطاء[a]	مطورو برمجيات إكسبلويت في BUSBY (2002-2016)
مع	281.97 يومًا = نحو 0.77 سنة (n=635)	1405.28 يومًا = نحو 3.85 سنوات (n=21)
بدون	912.45 يومًا = نحو 2.49 سنة (n = 284)	1639.33 يومًا = نحو 4.49 سنوات (n=18)

[a] كما هو مسجل في نشرات مايكروسوفت الأمنية، 1997-2016

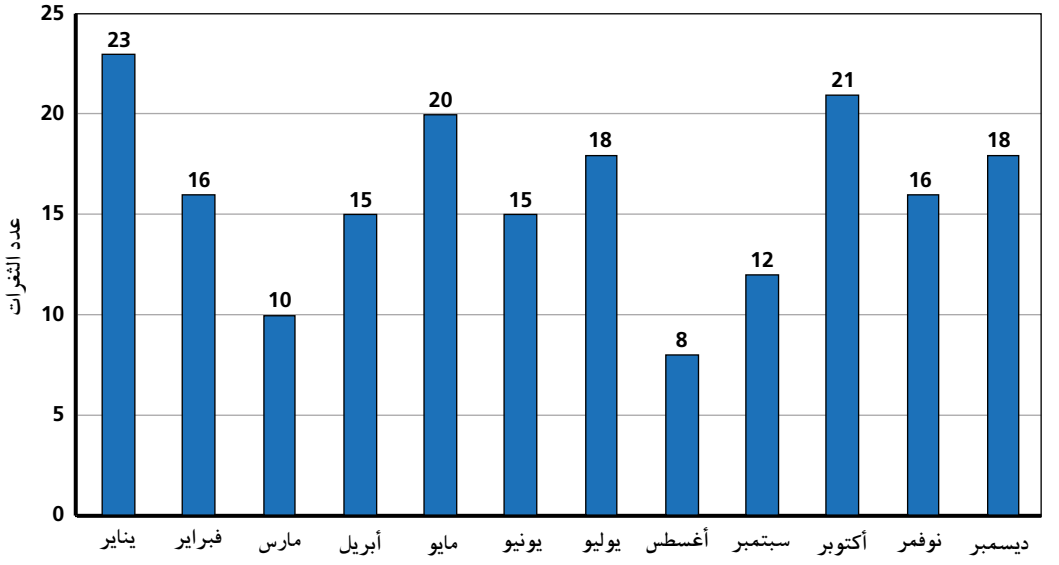
موسمية باحثي الثغرات

أردنا معرفة ما إذا كان اصطياد الأخطاء والبحث عن الثغرات عملًا موسميًا. انطلاقًا من هذا، هل هناك أشهر معينة خلال السنة يقل فيها اكتشاف ثغرات يوم الصفر التي تستحق الاستغلال أو يزداد؟ وهل يمكن اكتشاف ثغرات قابلة للاستغلال في أشهر معينة؟

باعتبار تحديد تاريخ "الميلاد" على أنه تاريخ اكتشاف باحثي الثغرات في BUSBY، قمنا بفصل الثغرات التي وجدناها بحسب الشهر (راجع الشكل B.3)

[5] إنها على الأرجح عينة غير موثوقة، بما أن بعض الباحثين قد يسعون إلى اكتشاف الثغرات في منتجات غير مايكروسوفت، وقد لا تكون مايكروسوفت سجلت جميع الثغرات التي أبلغت بها. وعلى نحو مماثل، قد لا تكون قيم مطوري برمجيات إكسبلويت ممثلة بما يكفي بما أنه لم يتم الإبلاغ عن بعض برمجيات إكسبلويت المطورة والثغرات المكتشفة أو تسجيلها في بياناتنا بسبب الحساسيات أو الاستعمال التشغيلي القائم عند جمع البيانات.

الشكل B.3
موسمية باحثي الثغرات: الأشهر التي وجد خلالها باحثو BUSBY للثغرات ثغرات يمكن استغلالها (n̄=21)



RAND *RR1751-B.3*

نلاحظ في هذه الحالة وجود عامل الموسمية. فيبدو أن النسبة الأدنى لاكتشاف ثغرات قابلة وجديرة بالاستغلال كانت خلال شهر آب (أغسطس)، في حين شهد شهر كانون الثاني (يناير) النسبة الأعلى لاكتشاف الثغرات.[6]

قد يتمثل أحد التفسيرات المحتملة في أن العديد من العاملين في BUSBY يكونون في عطلتهم الصيفية خلال هذا الشهر.[7] وفي حين أن شهر آذار (مارس) لم يسجل نتيجة مهمة من الناحية الإحصائية، إلا أنه أظهر تراجعًا. ويعتبر شهر كانون الثاني (يناير) مهمًا بالنسبة إلى صائدي الأخطاء. وربما تساهم قرارات السنة الجديدة في اكتشاف أخطاء قابلة للاستغلال في هذا الازدهار، أو قد يعني أنه خلال الأشهر الأكثر برودة يبقى عدد أكبر من الناس داخل منازلهم وليس لديهم ما يفعلونه سوى اكتشاف الثغرات القابلة للاستغلال. كما أن شهر آذار (مارس) يعتبر شهرًا بطيئًا بالنسبة إلى جميع أنواع المحللين. قد تكون عطلة الربيع أو بداية الفصل الأكثر دفئًا والأيام الأطول سببًا للتراجع.

[6] يعتبر العدد الضئيل للثغرات خلال آب (أغسطس) مهمًا من الناحية الإحصائية، في حين لا يكتسب العدد الكبير خلال كانون الثاني (يناير) أهمية كبيرة. تبلغ احتمالية أن يسجل أي شهر ثمانية اكتشافات أو أقل 20%. أما احتمال أن يكون هذا الشهر هو آب (أغسطس) مع الإبلاغ عن ثماني ثغرات أو أقل فهو أكثر أهمية، ويبلغ 2.4%. وفي ظل وجود 192 نتيجة، تعادل فرضيتنا الباطلة بأن كل اكتشاف يحدث في آب (أغسطس) احتمال 12/1. في هذه الحالة، نتوقع تسجيل نحو 16 اكتشافًا في آب (أغسطس) (12/192 = 16). غير أن هذا الشهر سجل ثمانية اكتشافات، بقيمة P = 0.0181. وتعتبر هذه القيمة المنخفضة جدًا دليلًا على أن الاحتمال الفعلي لحدوث اكتشاف في آب (أغسطس) هو أقل من 12/1. في المقابل، تبلغ قيمة P لـ23 اكتشافًا خلال كانون الثاني (يناير) 0.05034، ما يعني أنه من المستبعد أن يكون احتمال الاكتشاف خلال كانون الثاني (يناير) أكبر من 12/1.

[7] يتمثل شرح آخر في أنهم قد يكونون ذهبوا للمشاركة في المؤتمرات الأمنية على غرار بلاك هات (BlackHat) وديف كون (DEFCON).

الملحق C

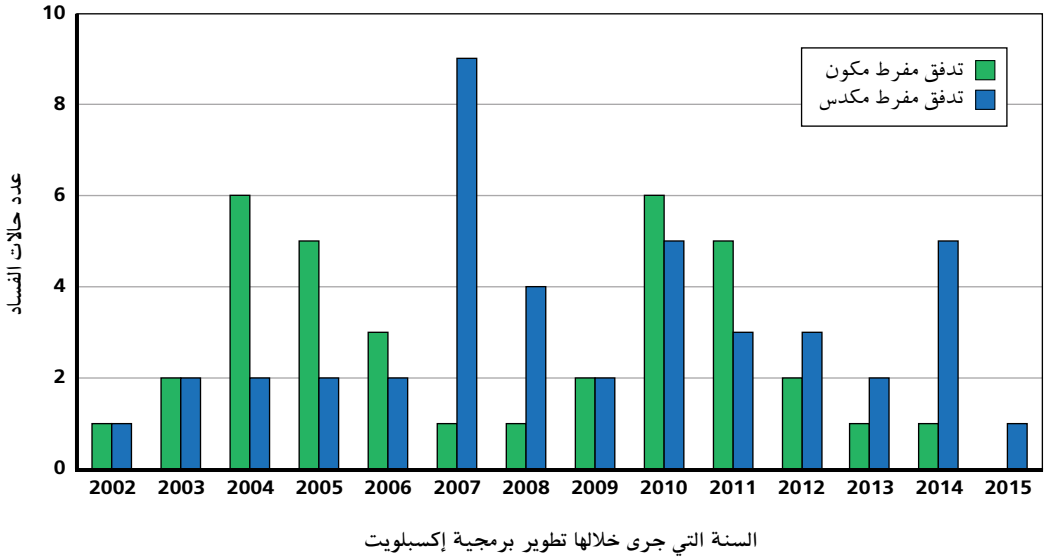
كيف أثرت عمليات التخفيف على قابلية الاستغلال: دراسة حالة الاستغلال في بنية البيانات المكدسة مقابل المكونة

لقد تبدلت قابلية استغلال ثغرات فساد الذاكرة أيضًا مع الوقت، لا سيما بين التكديس والتكوين. قبل عام 2007، كان استغلال التكديس أصعب من التكوين، ويعزى ذلك عمومًا إلى الفوارق من حيث التعقيد (ففهم التكوين والتلاعب به أبسط عادةً من التكديس).[1] ولهذا السبب، كانت برمجيات إكسبلويت القائمة على تدفق مفرط مكون أكثر شيوعًا، وعليه، ركز البائعون الأمنيون على استحداث عمليات تخفيف للتكوين (مثل ملفات ارتباط للتكوين والبلابل التي ترصد التكوين وتخطيط مساحة عنوان التوزيع العشوائي الخاص بالتكوين). وقد دفع هذا الأمر بمطوري برمجيات إكسبلويت إلى التركيز على التكديس، وهي مشكلة أصعب في جوهرها، لكنها تتطلب عمليات تخفيف أقل في الوقت نفسه، ما يؤدي إلى نجاح أفضل فور رصد مسارات الفساد. وبحلول 2010، برزت أساليب إضافية لحماية التكديس، ما أدى إلى انتفاء الانحياز.

ورغم أن ما جرى استغلاله (التكديس مقابل التكوين) تبدل مع الوقت، لم يتغير معدل تطوير ثغرات فساد الذاكرة (بما فيه الوقت اللازم لتطوير برمجية إكسبلويت)، كما لم يؤثر ذلك على الثغرات المكتشفة. ويظهر الشكل C.1 كيف تبدل مطورو برمجيات إكسبلويت في BUSBY مع تطوير برمجيات إكسبلويت لثغرات فساد الذاكرة.

[1] يعزى ذلك عمومًا إلى واقع أن التكديس يتمتع بعدة أنواع من الذاكرة التي قد تفسد، ويصعب معرفة العملية المتضررة. فحتى أن أبسط عملية تدفق مفرط مكدس تعني التعامل مع قوائم مرتبطة منفردة ومزدوجة، والعمل مع أجزاء من الرمز مرتبطة بطرق مختلفة (أي كيفية ترابط الأشياء بشكل منطقي وفي الواقع). أما فساد أو تدفق مفرط في التكوين فيكون أبسط بكثير؛ حيث إن التكوين يرتفع وينخفض، كما أن اختيار الخوارزمية لمعرفة الذاكرة التي يتم استعمالها أكثر بساطةً.

الشكل C.1
نوع فساد الذاكرة، العدد سنويًا (n=101)



RAND *RR1751-C.1*

الملحق D

اصطدامات قريبة

يقال إن الاصطدامات القريبة تحدث حين تكون الثغرات قريبة جدًا من بعضها البعض في مجال الرموز، ولكن يتم اكتشاف أحدها، ما يترك الثغرات المتبقية مجهولة للعامة وقابلة للاستغلال. أحيانًا، يجري اكتشاف ثغرات جديدة نتيجة مراقبة مجال الرمز المكتشف ضمن نشرة أو أنظمة الثغرات الشائعة والانكشاف، واكتشاف ثغرات جديدة. في أحيان أخرى، يتمّ إطلاق نظام ثغرات شائعة وانكشاف يكون قريبًا جدًا (أي ضمن مجال الرمز المشابه) إلى ثغرة قائمة أساسًا، لكن الثغرة القائمة لا تزال حية.

على سبيل المثال، عند اكتشاف أن ثغرات الخطوط كانت تستخدم في ستوكسنت، بدأ العديد من الباحثين الأمنيين بالبحث عنها، حيث قاموا بإبادة فئة كاملة من الثغرات بشكل أساسي. ويشعر بعض الباحثين الأمنيين بأنه حتى في حال وجود ثغرات خطوط، ليس مجديًا من الناحية التشغيلية اكتشافها واستغلالها، بما أن العديد من الباحثين يسعون إليها ولن تبقى الثغرة على الأرجح مجهولة للعامة لفترة طويلة.

وثمة مثال آخر حيث تم اكتشاف إحدى برمجيات إكسبلويت تابعة لـBUSBY بفضل مطور برمجيات إكسبلويت قام بمراقبة تفاصيل نشرة ثغرات تمّ إصدارها حديثًا، واكتشاف ثغرات مشابهة أخرى، ولا تزال مجهولة للعامة (أي حية).

يذكر أن 24 برمجية من برمجيات إكسبلويت BUSBY تعتبر اصطدامات قريبة. واعتبارًا من تاريخ توقف بياناتنا (1 آذار (مارس)، 2016)، كان أكثر من الثلث بقليل (9) لا تزال حية (1 هي عامل إعادة هيكلة الرمز)، رغم إصدار إصلاحات أمنية وأنظمة الثغرات الشائعة والانكشاف الخاصة بالثغرات الموجودة ضمن مجال الرمز نفسه. تجدر الملاحظة أن أكثر من النصف (13) ميتة (من خلال إصلاح أمني).[1]

وقد تحدث الاصطدامات القريبة حتى ضمن الشركة الواحدة: فقد كانت خمس برمجيات إكسبلويت من BUSBY متداخلة مع سواها. وكانت ثلاث منها خاصة بالثغرة نفسها، إنما تتمتع بخوارزميات مختلفة، ولم يعرف أنها نفسها إلا عند إصدار نظام الثغرات الشائعة والانكشاف/ النشرة.[2] وقد اكتشف المحلل نفسه (D) ثلاث منها. وكانت البرمجيتان الأخريان هما الثغرة نفسها، واحدة لأنظمة 32 بت وأخرى لأنظمة 64 بت، وقد جرى اكتشاف واستغلال كل منها على حدة بفارق ثلاث سنوات (2005 و2008)، علمًا بأن المحلل (P) نفسه اكتشفهما.

[1] أما الاثنتان المتبقيتان فحالتهما غير مؤكدة لمعرفة إن كانتا لا تزالان حيتين.

[2] جرى اكتشاف اثنتان واستغلالهما مع فارق شهر بينهما، في حين اكتشفت الثالثة بعد أكثر من عام.

الملحق E

شراء برمجية إكسبلويت لاستغلال ثغرة يوم صفر: بعض التكاليف والاعتبارات السعرية

في هذا الملحق، نسعى إلى منح قيمة نقدية لاكتشاف ثغرة وتطوير برمجية إكسبلويت. ونركز فقط على الوقت اللازم لاكتشاف ثغرة (وقت البحث)، والوقت اللازم لتطوير برمجية إكسبلويت (وقت تطوير برمجية إكسبلويت) والراتب باعتباره مقياسًا للتكلفة. وحتمًا، قد تدخل جوانب أخرى خاصة بالعرض والطلب في قيمة برمجية إكسبلويت (كما ذُكر في الفصل الثالث). وعليه، يجب استخدام هذا التحليل باعتباره نقطة انطلاق لصقل التكلفة الضرورية لتطوير برمجية إكسبلويت.

الوقت اللازم لاكتشاف الثغرات

قد يختلف الوقت اللازم لاكتشاف ثغرة استنادًا إلى الإعداد، والجزء المحدد من الرمز المراقب وعمر الرمز المراقب. غير أنه لم يتمّ تسجيل وقت هذا البحث ضمن بياناتنا، لكننا تحدثنا إلى ستة فرق أو شركات أو أفراد مختلفين سبقوا لهم أن عملوا أو شاركوا في تطوير برمجيات إكسبلويت للحصول على مساهمتهم المباشرة.[1] وقد أعلنت إحدى شركات تطوير برمجيات إكسبلويت أنه نادرًا ما كان وقت البحث يتخطى الشهرين وأن متوسط فترة البحث ناهز الشهر. في حين قدرت شركة أخرى أن فترة التدقيق استغرقت بين أسبوعين وثلاثة أسابيع.

وبالنسبة إلى الذين يستحدثون برمجيات إكسبلويت، يجري اكتشاف الثغرات عادة وفق مبدأ الحاجة، سواء لأن العميل طلب برمجية إكسبلويت أو أن مطور برمجيات إكسبلويت بحاجة إلى "تعبئة المكان" (أي كان المطور يملك ثلاث ثغرات نواة من لينكس لكن جرى إصلاح اثنتين، لذا عليه إيجاد المزيد ليتمكن من الانطلاق).

وتعتبر الفترة الأطول خلال البحث عن الثغرات هي فترة الإعداد مثل ضمان ملاءمة الاختبار العشوائي، وهي أداة لتحليل الرمز، للتطبيق الذي يختبره، مع الحرص على صحة نماذج الملف وهندسته، أو إعداد التشكيلات والبنية التحتية المناسبة. وبالنسبة إلى من يجري

[1] شمل ذلك فريقين من منظمات مطوري برمجيات إكسبلويت ومجموعة من شركة تجارية كبيرة تقوم بصياغة برمجية إكسبلويت لأغراض الاختبار الداخلي، ومديرًا سابقًا لمقاول في مجال الدفاع يعنى بتطوير برمجيات إكسبلويت، وباحث ثغرات مرموق من أصحاب القبعات البيضاء ووسيط ثغرات سابق.

الفحص العشوائي يمكن، فور إعداد أداة الفحص، اكتشاف الثغرات بسرعة حتى أن أحدهم قدر أن يستغرق الأمر ثوانٍ، ويعزى ذلك عمومًا إلى أن الثغرات التي يكتشفها هذا الاختبار تكون عمومًا على السطح.[2] وقد يستغرق التدقيق في الرمز أو مراجعته يدويًا وقتًا أطول، إما أيامًا أو أسابيع أو أحيانًا أشهر، غالبًا لأن الباحثين يغوصون عميقًا في قاعدة الرموز.

ويمكن أن يعتمد الوقت الضروري لاكتشاف ثغرة على الجزء الذي يخضع للمراقبة من الرمز،[3] والمنتج الذي ستتم مراقبته، وسواء يمكن استعمال أدوات أو تقنيات سابقة لرصد الثغرات في منتجات جديدة،[4] وما هي عمليات التخفيف التي يمكن استخدامها على المنتج.[5] وعادة تتركز الثغرات التي تكتشف في أسرع وقت في برمجيات أحدث، ذات أهداف أصغر أو في برمجيات غير شائعة إلى حدّ كبير، لأن الثغرات التي يسهل اكتشافها لم تكتشف بعد.

وفي معظم الأحيان، سيمضي الباحثون عن الثغرات بعض الوقت (تشير التقديرات إلى فترة تتراوح بين أسبوع وشهر) في البحث عن ثغرات قابلة للاستغلال في منتج ما، إن لم يكتشفوا شيئًا مجديًا، يتركون المشروع جانبًا ويعاودون العمل عليه لاحقًا، أو قد يمررونه إلى أحد زملائهم ليلقي نظرة جديدة. وفي ظل وجود بعض الاستثناءات،[6] قد تجري عملية الانتقال مرتين أو ثلاث مرات (لمدة أقصاها 3 أشهر). قبل وضع المنتج جانبًا وانتقال الباحثين إلى شيء آخر.

الوقت الضروري لتطوير برمجية إكسبلويت

كنا ذكرنا في الفصل الثالث، لقد وجدنا أنه في حين يختلف الوقت الضروري لتطوير برمجية إكسبلويت، استغرق تطوير غالبية برمجيات إكسبلويت في مجموعة بياناتنا بين 6 و37 يومًا (مع متوسط 22 يومًا وحدّ أدنى بلغ يومًا واحدًا وحدّ أقصى بلغ 955 يومًا).

[2] في إحدى الدراسات، وجد الباحثون أن الفحوصات العشوائية اكتشفت أقل من 2% من الأخطاء في منتج أُدخلت إليه الثغرات (كيردا وآخرون، غير مؤرخ الشريحة 24).

[3] مثلًا، غالبًا ما قد يستغرق اكتشاف ثغرة في النواة تدقيقًا لمدة تتراوح بين 3 و4 أسابيع قبل اكتشاف ثغرة قابلة للاستغلال.

[4] مثلًا، استخدام الأدوات أو التقنيات نفسها في مختلف المتصفحات المفتوحة المصدر.

[5] مثلًا، أداة تجربة التّخفيف المحسّنة من مايكروسوفت أو ميزة حراسة انسياب التحكم، ما يصعب عمل الرمز في الأنظمة التشغيلية أو المتصفحات. ولتجنب هذين النوعين من عمليات التخفيف، قد يحتاج مطورو برمجيات إكسبلويت إلى استعمال أو استحداث سبل تطوير أخرى. وهو وقت يضاف إلى وقت البحث.

[6] يتمثل أحد الاستثناءات في أنه إن طلب عميل برمجية إكسبلويت لمنتج محدد وطلب من المطورين مواصلة البحث عنه. حتى في هذه الحالة لا يكون البحث متواصلًا. إن لم يجد مطورو برمجيات إكسبلويت شيئًا بعد بضعة أشهر من العمل (وانتقال المشروع بين مجموعة محدودة من مطوري برمجيات إكسبلويت)، قد ينتظرون إلى حين صدور نسخة جديدة عن المنتج ويخصصون أسبوعًا للبحث عن ثغرات جديدة أو مسارات رموز لثغرات كان يستحيل الوصول إليها من قبل. أما استثناء آخر، فهو أنه يجري توظيف مطوري برمجيات إكسبلويت ليبذلوا جهودًا طويلة الأمد في منصة (مثلًا نوع محدد من الآلات الافتراضية). ويمكن إبرام عقود معهم لتمضية نصف عام في اكتشاف الثغرات. وقد يؤدي هذا النوع من الجهود الطويلة الأمد إلى اكتشاف العديد من الثغرات، يجري استغلال القليل منها فقط، في حين يوضع الباقي جانبًا للاستعمال المستقبلي في حال اكتشاف مجموعة الثغرات في برمجية إكسبلويت وإصلاحها. ونقدم مثالًا آخر أيضًا على تمديد وقت البحث ألا وهو في حال كان المنتج مهمًا لنموذج عمل شركة (توفير محفز للباحثين المحليين ليركزوا عليه أطول فترة ممكنة). وقد لفتت إحدى المجموعات إلى أنها كانت تدرس منتجًا لمدة 18 شهرًا.

تكلفة التحليل: كم تبلغ تكلفة تطوير برمجية إكسبلويت؟

من خلال التركيز على راتب المطور وحده، نستخدم تقديراتنا للوقت الضروري لاكتشاف ثغرة والبيانات اللازمة في الموعد المناسب من أجل تطوير برمجية إكسبلويت لتحديد أدنى قيمة تكلفة التطوير.

الأجر الأساسي لمطور برمجيات إكسبلويت: سنبدأ بمتوسط الأجر السنوي لمطور برمجيات إكسبلويت x، ويترجم ذلك بتكلفة 1.4x لصاحب عمل.[7] وباستخدام متوسط 261 يوم عمل في السنة (مكتب إدارة شؤون الموظفين الأمريكي، 2016)، يبلغ متوسط المعدل اليومي لمطوري برمجيات إكسبلويت الموظفين بدوام كامل (1.4/261) x.

تكلفة اكتشاف ثغرة: وفق التقديرات التي حصلنا عليها من مختلف شركات تطوير برمجيات إكسبلويت، تراوح الوقت اللازم لاكتشاف ثغرة بين أسبوعين وثلاثة أشهر، مع متوسط ناهز شهرًا واحدًا.

تكلفة تطوير برمجية إكسبلويت، بعد اكتشاف الثغرة: يختلف الوقت اللازم لتطوير برمجية إكسبلويت، لكن تطوير غالبية برمجيات إكسبلويت في مجموعة بياناتنا استغرق بين 6 و37 يومًا (مع متوسط 22 يومًا، حدّ أدنى يوم واحد وحدّ أقصى 955 يومًا).

باعتبار أن متوسط راتب مطور برمجيات إكسبلويت يبلغ 150 ألف دولار،[8] 40% منها تكاليف عامة لصاحب العمل، يبلغ متوسط الوقت الضروري لاكتشاف ثغرة (30 يومًا، أي نحو 21.45 يوم عمل)، ومتوسط الوقت الضروري لتطوير برمجية إكسبلويت (22 يومًا، أي نحو 15.73 يوم عمل)، مع أخذ أجر مطور برمجية إكسبلويت فقط في الحسبان، **علمًا بأن التقديرات تشير إلى أن متوسط إجمالي تكلفة العمالة لبرمجية إكسبلويت يناهز 29.914.95 دولارًا**.

وببساطة، يمكن لأي مهاجم خطير الحصول على ثغرة يوم صفر بسعر ميسور لأي هدف تقريبًا. غير أن تكاليف ملموسة أخرى (الاستحواذ على منتجات لإيجاد ثغرات فيها وإنشاء بنية تحتية للاختبار واستئجار مكان عمل، الخ) وتكاليف غير ملموسة (قيمة الطلب المرتفع ومنتج محدود العرض، الخ) قد تتسبب بارتفاع الأسعار بشكل ملحوظ. حرصًا على ضمان الوضوح، لا يأخذ تحليلنا في الحسبان تكلفة أو وقت اكتشاف ثغرة أو تطوير برمجية إكسبلويت فعالة بالكامل. علاوةً على ذلك، نظرًا إلى فارق الوقت الكبير بين اكتشاف الثغرات وتطوير برمجيات إكسبلويت، يمكن أن يختلف المدى الفعلي للتكلفة بشكل حاد.

[7] تتراوح تقديرات التكلفة الإضافية لموظف بالنسبة إلى صاحب العمل بين 18 و40%. (راجع salary.com، 2016؛ باغليري Pagliery، 2013؛ هادزيما Hadzima، 2005). نستخدم نسبة 40% باعتبارها تقديرًا متحفظًا، خاصة أنها تأخذ في الحسبان الفوائد الواضحة على غرار المجال المادي والتأمين الصحي. كما قد تبرز فوائد أخرى (على غرار خياري الطعام المجاني أو عطلة طويلة) من شأنها رفعها أكثر.

[8] قمنا بدراسة إعلانات الوظائف عبر الإنترنت وحصلنا على نطاقات من مختلف مطوري برمجيات إكسبلويت والباحثين عن الثغرات. كما أبلغنا أحد المطورين أن مطوري برمجيات إكسبلويت الصغار قد يتقاضون حوالى ٥٠ ألف دولار سنويًا، في حين تتراوح أجور المطورين المتمرسين غالبًا بين 120 و140 ألف دولار.

التسعير للبيع

كما ذكرنا سابقًا، تختلف الجهود المبذولة ومستويات المهارات المستخدمة لاكتشاف مختلف الثغرات واستغلالها. ويعتقد البعض أن تكلفة برمجية إكسبلويت يجب أن تزيد للثغرات التي يصعب اكتشافها واستغلالها (مثلًا تسخير حالة سباق)، في حين يرى آخرون أنه يجب إيلاء الأهمية لأثر الثغرات وخطورتها لتحديد التكلفة.[9] ويتمثل عامل آخر في انتشار المنتج الذي اكتُشفت فيه الثغرة وحصته السوقية.

تتمتع مختلف الأسواق على الأرجح بمخططات دفع متنوعة. فقد صرح البعض أن الأسعار في الأسواق الرمادية (أو الحكومية) والسوداء تكون أكثر بعشر مرات من تلك المعروضة في السوق الأبيض، ويعزى ذلك عمومًا إلى أن السوقين الرمادي والأسود غالبًا ما يعملان ببرمجيات إكسبلويت فعالة بالكامل وموثوقة، في حين أن برمجيات إكسبلويت في السوق الأبيض غالبًا ما تريد معلومات عن الثغرة فقط (ليبيكي وأبلون وويب، 2015). ورغم وجود استثناءات قليلة، لا يتقاطع السوقان الأبيض والرمادي، لأن حوافز كلا منهما مختلفة عمومًا.

تجدر الملاحظة أنه يصعب الحصول على أسعار ثغرات يوم الصفر وبرمجيات إكسبلويت في الأسواق الرمادية (السوداء). وكما قال أحد الوسطاء، "القاعدة الأولى في أعمال ثغرات يوم الصفر هي عدم مناقشة الأسعار علنًا" (مقتبس عن غرينبرغ (Greenberg)، 2015)، لكن يمكن اكتشاف بعض المعلومات المحدودة علنًا من خلال منشورات محدودة أو تسريبات أو في بعض الحالات أخبار متداولة.[10] لكن مع ذلك قد تتغير الأسعار. وقد أشارت إحدى شركات البحث عن الثغرات التي تحدثنا إليها إلى أن أسعارها لبرمجيات إكسبلويت تفوق بواقع 3 إلى 5 مرات تلك الواردة في قائمة أسعار شركة زيروديوم (زيروديوم Zerodium، 2016).[11] هذا وقد زعم شخص مطلع على شراء ثغرات يوم الصفر وبيعها إلى أن أسعار برمجيات إكسبلويت في السوق الرمادي لنسخة تور براوزر باندل (Tor Browser Bundle) كانت أعلى بمقدارين أسيين من أسعار السوق الأبيض لثغرات فايرفوكس (Firefox). كما لفت أشخاص آخرون تحدثنا إليهم إلى أنه من الشائع أن يتقاضى مطورو برمجيات إكسبلويت والوسطاء مبالغ ضخمة من أي شخص قد يقوم بنشرها، في مسعى إلى فرض وتيرة العمل في السوق. وفي حين أن برمجيات إكسبلويت "يونيكورن" (unicorn) موجودة، على غرار برمجيات إكسبلويت الكاملة السلسلة في أجهزة آي فون، إلا أنها ليست القاعدة، ونادرًا ما تصل إلى عتبة المليون دولار. عوضًا عن ذلك تباع معظم برمجيات إكسبلويت في السوق الرمادي أو الحكومي لقاء مبلغ يتراوح بين 50 و100 ألف دولار، وقد تصل إلى ما بين 150 و300 ألف دولار، بحسب كل برمجية. ويمكن مقارنة ذلك

[9] انطلاقًا من هذا، وجد ألودي (Allodi) (2014) أن الدرجة ضمن نظام تصنيف الثغرات المشتركة قد تكون دليلًا على الاستعمال في العلن، علمًا بأن درجات هذا النظام أو درجة الخطورة وفق أنظمة الثغرات الشائعة والانكشاف قد لا تكون دقيقة.

[10] مثلًا، كتب ميلر (Miller) (2007) ورقة حملت بعض الرؤى الأولية لأسعار ثغرة يوم الصفر. وقد تحدثت غرينبرغ (Greenberg) (2012) عن أسعار برمجيات إكسبلويت لاستغلال ثغرات يوم الصفر، كما فعلت عدة وكالات إخبارية). بدورها، تحدثت رويترز (Reuters) عن بيع ثغرة يوم صفر إلى مكتب التحقيقات الفيدرالي (هوسنبول Hosenball، 2016). وقد كشفت تسريبات الرسائل الإلكترونية الخاصة بـ"هاكينغ تيم" (Hacking Team) أسعار عدة ثغرات يوم صفر (تسيركليفش (Tsyrklevich)، 2015). وورد في مقال صحفي أن الحكومة الأمريكية أنفقت 25 مليون دولار في 2013 على برمجيات إكسبلويت (غيلمان Gellman وناكاشيما Nakashima، 2013).

[11] زيروديوم (Zerodium) هي شركة متخصصة في اكتشاف ثغرات يوم الصفر وبيعها إلى الحكومات والشركات. وقد أسسها الرئيس التنفيذي لشركة فوبين (VUPEN) التي كانت تعنى بالسوق الرمادي ومتخصصة في اكتشاف ثغرات يوم الصفر وبيعها إلى الوكالات الحكومية.

مع برمجيات إكسبلويت في السوق الأسود التي يكون سعرها أقل، فبرمجية إكسبلويت لاستغلال الثغرات في برمجية "فلاش" (Flash) قد تباع بسعر يتراوح بين 30 و50 ألف دولار.

أما أسعار ثغرات يوم الصفر في السوق الأبيض (مثل المكافأة الأمنية) فهي متوافرة بسهولة أكبر. ويتمتع كل برنامج مكافآت أو شركة بسبل مختلفة بعض الشيء لصياغة المدفوعات أو عرضها، كما أن طريقة تحديد هذه المدفوعات تختلف بحسب عدد من العوامل. وقد نشر البعض أوراقًا أو مقالات عن التصنيف أو الوسيلة المعتمدة لتحديد المدفوعات، إلى جانب المبالغ المدفوعة إلى كل شخص.[12] وتتراوح مدفوعات الثغرات في المتوسط بين 300 و650 دولارًا لكل ثغرة من خلال منصات المكافأة الأمنية وقد تصل إلى الآلاف أو مئات الآلاف، بحسب كل منظمة أو شركة على حدة.[13]

وبشكل عام، بالنسبة إلى أي سوق، تعزى هيكلية الدفع إلى عدد قليل من العوامل تشمل: درجة سهولة أو صعوبة اكتشاف الثغرة (إن كانت صعبة فسيكون المبلغ أكبر)، وكم ثغرة أخرى تمّ اكتشافها في المنتج (إن كان عددها قليلًا، قد تكون الثغرات متناثرة وبالتالي سيكون المبلغ أكبر)، والأثر (إن كانت برمجية إكسبلويت قادرة على القيام بأكثر من مجرد تنفيذ الرمز عن بعد، بل تستطيع أيضًا تجنب عملية التخفيف أو التهرب من آلية تحديد الوصول أو كسر الحماية عن بعد [للأجهزة المحمولة]، سيكون المبلغ أكبر).

في نهاية المطاف، بالنسبة إلى الذين يبيعون برمجيات إكسبلويت الخاصة بهم، قد تكون الكلمة الفصل غالبًا للجهة التي تشتري الثغرة بشأن ما يجب شراؤه ولقاء أي سعر، بغض النظر عن الوقت اللازم لاكتشاف (أو استغلال) الثغرة أو نوعها.

---

[12] مثلًا، أصدرت باغ كراود (Bugcrowd) نموذجها لتسعير الثغرات الدفاعية، حيث يقوم تسعير الثغرات على عدد من العوامل، بما فيها أثر الثغرة والنضوج الأمني لمنظمة (باغ كراود، 2015). من جهتها، طورت سيناك نظام تصنيف خاص بها، حيث تحدد مبالغ المكافأة الأمنية بفضل الاكتشاف وتصنيف الخطر (شيبير Shieber، 2014). أما هاكر وان (HackerOne)، فقد نشرت المبادئ التوجيهية لمدفوعات الثغرات للشركات التي تأمل إطلاق برنامج المكافأة الأمنية الخاص بها (هاكر وان، 2016). يذكر أن مايكروسوفت تتمتع "بميزة شرفية" على صعيد صائدي المكافأة الأمنية (مركز مايكروسوفت للتكنولوجيا، 2016a).

[13] أعلنت هاكر وان (HackerOne) أن الباحثين الذين يستخدمون منصتها يكسبون في المتوسط 650 دولارًا عن كل عيب مكتشف (خاربال Kharpal، 2015)، كما ذكرت أن متوسط مدفوعاتها لكل خطأ بلغ 294.70 دولارًا (باغ كراود، 2016). وخلال مبادرة وزارة الدفاع الأمريكية التي حملت اسم "قرصنة البنتاغون" (Hack the Pentagon) تقاضى الباحثون بين 100 و15 ألف دولار (فيرديناندو Ferdinando، 2016). وقد وصل بدل الأتعاب الأقصى خلال برنامج المكافأة الأمنية الخاص بآبل (المعلن عنه في آب (أغسطس) 2016) إلى 200 ألف دولار (فينكل Finkle، 2016). وقد قدمت مايكروسوفت جائزة كبرى بقيمة 100 ألف دولار لتقنيات الحدّ من التجاوز (مركز مايكروسوفت للتكنولوجيا، 2016b).

أشكال وجداول إضافية

يوفر هذا الملحق أشكالاً وجداول ورسومًا بيانية إضافية تمّت الإشارة إليها خلال هذا التقرير.

الجدول F.1
تواتر الخصائص على مستوى برمجيات إكسبلويت
من بين 127 برمجية محددة

الخاصية	N (%)
نوع الثغرة	
منطقية	36 (28.4%)
فساد الذاكرة	64 (50.4%)
سوء إدارة الذاكرة	20 (15.8%)
مختلطة	6 (4.7%)
غير ذلك	1 (0.8%)
المنصة المتضررة	
لينكس	33 (26%)
مختلطة	4 (3.2%)
ماك أو.إس	8 (6.3%)
مفتوحة المصدر	13 (10.2%)
غير ذلك	3 (2.4%)
بي إتش بي	8 (6.3%)
قائمة على يونكس	10 (7.9%)
ويندوز	48 (37.8%)
نوع المصدر	
مفتوح	50 (39.4%)
مغلق	70 (55.1%)
مختلط	1 (0.8%)
مجهول	6 (4.7%)
نوع الفئة	
من جانب العميل	12 (9.5%)
محلية	50 (39.4%)
مختلطة	12 (9.5%)
غير ذلك	10 (7.9%)
عن بعد	43 (33.9%)

يظهر الجدول F.2 عدد برمجيات إكسبلويت التي لا تزال حية في بداية فواصل زمنية لعام واحد، وعدد تلك التي من المتوقع أن تموت وعدد المفقود منها من أجل المتابعة بسبب الحظر المفروض (أي في نهاية الوقت الذي استخدمناه عينة بين 2002 و2016، هذه هي الثغرات التي كانت لا تزال حية، لكن لم نتمكن من ذكر أي معلومات حول احتمال بقائها).

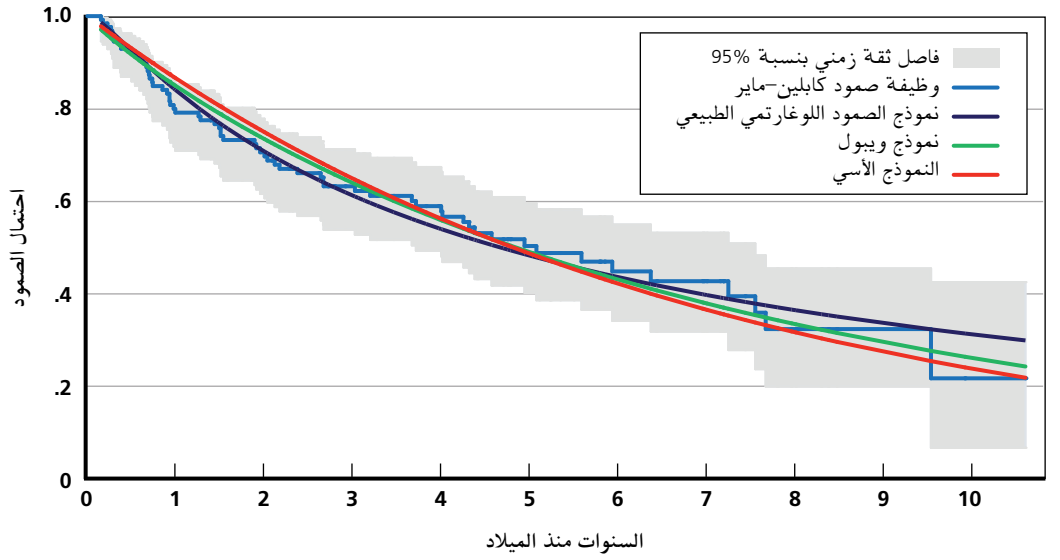
الجدول F.2
جدول الحياة، جميع البيانات

الفاصل الزمني في بداية العام	الفاصل الزمني في نهاية العام	مجموع البدء	حالات الموت	حالات الحظر	احتمال البقاء (مجال ثقة 95%)
0	1	127	26	5	0.79 (0.71, 0.85)
1	2	96	10	7	0.71 (0.62, 0.78)
2	3	79	8	9	0.63 (0.54, 0.71)
3	4	62	5	7	0.58 (0.48, 0.66)
4	5	50	6	10	0.50 (0.40, 0.59)
5	6	34	3	10	0.45 (0.34, 0.55)
6	7	21	1	3	0.42 (0.32, 0.53)
7	8	17	3	7	0.33 (0.21, 0.46)
8	9	7	0	4	0.33 (0.21, 0.46)
9	10	3	1	1	0.20 (0.04, 0.44)
10	11	1	0	1	0.20 (0.04, 0.44)

يظهر الشكل F.1 النموذج الأسي و نموذج وايبول (Weibull) ونموذج الصمود اللوغارتمي الطبيعي بما أنها تلائم بياناتنا من خلال مقارنة مخططاتها المقدرة للصمود مع مخططات كابلين–ماير. وتكمن النماذج البارامترية ضمن شريط الثقة حول منحنى كابلين–ماير، رغم أنها تميل إلى المبالغة في تقدير هذا المنحنى خلال العامين الأولين. وبالتالي، يقترب كل منهما من الشكل العام لمنحنى كابلين–ماير، مع معدلات اندثار متفاوتة بدءًا من العام الخامس. لقد اخترنا النموذج الأسي المحدد باللون الأحمر. وقد استخدمنا إطار الاندثار من النموذج الأسي للتخطيط لمنحنيات الصمود الأسي (أي حيث يبلغ المنحنى في نهاية المطاف أو يقترب كثيرًا من قدرة صمود صفرية – وهو أمر عجزنا عن فعله مع البيانات وحدها) إلى جانب فواصل زمنية للثقة بنسبة 95% من إجمالي البيانات، فضلًا عن خصائص محددة لبرمجيات إكسبلويت.

أما الشكل F.2، فيظهر وظيفة الخطورة السلسة التي تناسب مخطط جدول الحياة. وتشير القيم الأعلى إلى الأماكن التي يكون فيها خطر الموت هو الأعلى.

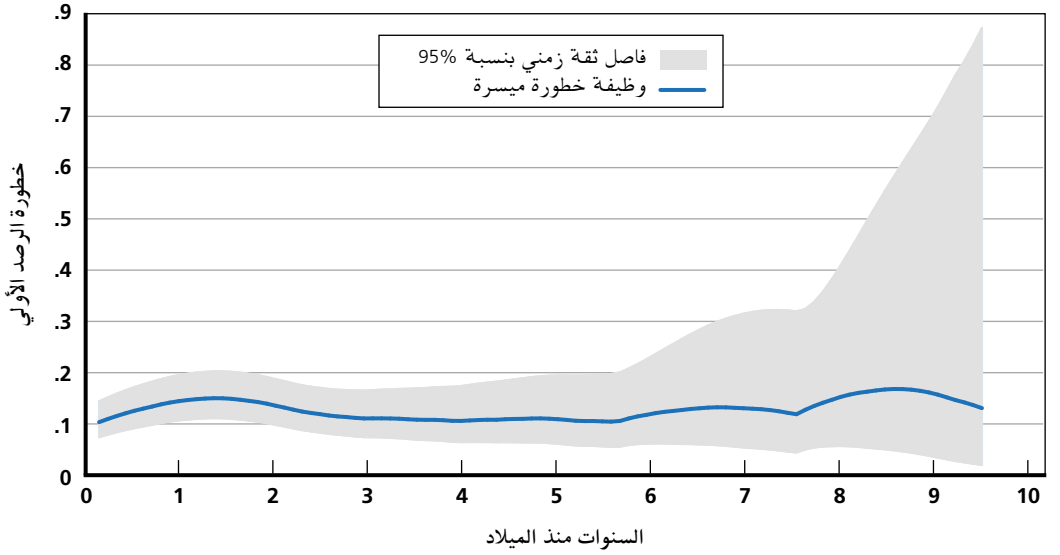
الشكل F.1
مقارنة نماذج بارامترية مع كابلين–ماير (n=127)



RAND *RR1751-F.1*

الشكل F.2
وظيفة خطورة ميسرة

الوقت منذ الميلاد (الرصد الأولي من قبل مطور برمجيات إكسبلويت) للرصد من قبل طرف خارجي



RAND *RR1751-F.2*

ويكون شريط الثقة عند 95% الذي يبرز عند كل نقطة مشمولاً ليساعد على أخذ هذه الارتفاعات والانخفاضات في الحسبان: وكما يظهر في الشكل F.2، يكون شريط الثقة عريضًا نسبةً إلى أحجام الارتفاعات والانخفاضات، ما يشير إلى أن الخطورة مستمرة مع مرور الوقت إلى حدّ ما، وما من دليل قوي على أن عددًا محددًا من السنوات منذ الميلاد يشكل خطورة أكبر بالنسبة إلى حالات الموت من غيرها. وكلما نظرنا بشكل أكبر إلى الوقت، اتسع نطاق الثقة وتراجعت ثقتنا بالشكل الحقيقي لوظيفة الخطورة. وواقع أن الخطورة شبه مستمرة يعتبر مبررًا جيدًا لاستعمال نموذج الصمود الأسي من أجل تقدير أمد الحياة المتوقع، بما أن النموذج الأسي قائم على فرضية الخطورة المستمرة. وفي حالتنا، تبلغ قيمة الخطورة 0.145 (إن فاصل الثقة الزمني عند 95% يبلغ: 0.113، 0.186)، ما يبدو نظريًا ممتدًا على طول الممر الأفقي في الشكل F.2. وقد استخدمنا قيمة الخطورة عند 0.145، إلى جانب حدود الثقة الخاصة بها، من أجل إصدار تقديرات النموذج الأسي لأمد الحياة المتوقع.

لقد استخدمنا نموذج كوكس لانحسار الخطورة النسبية من أجل تحديد معدل الخطورة لكل خاصية.

ويمكن تفسير وظيفة الخطورة على أنها الاحتمال الفوري للموت في أي وقت محدد t، بافتراض عدم اكتشاف ثغرة عند بلوغ هذه المرحلة من الوقت. ويعتبر معدل الخطورة المترافق مع خاصية محددة الكسر الناتج عن قسمة الخطورة لمجموعة واحدة في الوقت t على تلك الخاصة بمجموعة مقارنة (أو مرجع) في الوقت نفسه. ومن حيث المفهوم، يشبه ذلك خطرًا نسبيًا، حيث يشير رقم كبير إلى خطر أكبر يترتب عن حادثة بالمقارنة مع مجموعة مرجعية، كما أن عددًا أدنى يشير إلى خطر أقل. ويشير معدل خطورة إحداهما إلى أن احتمال فشل أعضاء المجموعتين في أي وقت متساوٍ. على سبيل المثال، إن كان معدل خطورة الفشل لفساد الذاكرة عند سنتين يبلغ 0.14، في حين يبلغ معدل خطورة فشل ثغرات سوء إدارة الذاكرة 0.11، عندها يبلغ معدل الخطورة 1.27. علاوةً على ذلك، باستخدام نموذج كوكس للخطورة النسبية، نفترض أن هذه العلاقة بين وظائف الخطورة ثابتة مع مرور الوقت، وقد ترتفع خطورة الفشل أو تنخفض بالنسبة إلى هذين النوعين من الثغرات، لكن المعدل بينهما ثابت مع مرور الوقت. ودائمًا ما تكون خطورة فشل ثغرات فساد الذاكرة 1.27 مرة أكبر من ثغرات سوء إدارة الذاكرة. وتسمح لنا إجراءات النمذجة أيضًا باختبار هذا المعدل نظرًا لأهميته الإحصائية، حيث نقوم بالمقارنة مع قيمة فرضية باطلة تصل إلى 1.0، ما يعني أن وظائف الخطورة متعادلة دائمًا.

وبالنسبة إلى كل خاصية، اخترنا فئة واحدة لتكون مرجعية، شرط أن تكون وظيفة الخطورة الخاصة بها القاسم لمعدلات الخطورة على غرار تلك التي تم وصفها أعلاه.[1] يذكر أنه جرى حساب ثلاثة نماذج مختلفة للحصول على:

1. معدل خطورة غير معدل
2. معدل خطورة معدل بالكامل

[1] يكون من المفيد غالبًا اختيار مجموعة مرجعية تتمتع بعدد كبير من السجلات، للحصول على أكبر قدرة ممكنة لاختبارات كل معدل خطورة فردية (بالمقارنة مع القيمة الفرضية الباطلة عند 1، ما يعني أن معدلات الخطورة متعادلة). لكن من وجهة نظر علمية، يمكن للمرء أن يختار أيضًا النوع الذي يعتبر "الأكثر طبيعية" لفهم التقديرات الخاصة بهذا النوع. واخترنا مزيجًا من الاثنين (عدد كبير من السجلات و"الأكثر طبيعية").

3. معدل خطورة معدل جزئيًا

في النموذج غير المعدل، جرى تقييم كل خاصية (نوع الثغرة والمنصة المتضررة ونوع المصدر ونوع الفئة) على حدة، لمعرفة ما إذا كانت أي سمة ضمن هذه الخصائص تشير إلى حياة طويلة أو قصيرة. وقد ظهرت القيم p الموجودة هنا في الرسوم البيانية في الشكل 3.7.[2] أما في النموذج المعدل بالكامل، فقد جرى تعديل كل خاصية بالتزامن ضمن نموذج شائع. وفي النموذج المعدل جزئيًا، ألغينا فئة "المنصة المتضررة"، لأن نوع المصدر مرتبط إلى حد كبير بالمنصة المتضررة (على سبيل المثال جميع النوافذ تكون مغلقة المصدر) ونفذنا التعديل الانحساري لنوع الثغرة ونوع المصدر ونوع الفئة. وكلما ارتفع معدل الخطورة، ازدادت خطورة الموت أو احتمال رصدها، بالمقارنة مع الفئة المرجعية. على سبيل المثال، من خلال مراقبة نوع فئة برمجية إكسبلويت، في أي وقت تكون فيه برمجيات إكسبلويت لا تزال حية ومجهولة للعامة، اكتشفنا أن احتمال موت برمجية إكسبلويت من جانب العميل يفوق بواقع 1.02 مرة برمجية إكسبلويت عن بعد، حيث إن نسبة اكتشاف برمجية إكسبلويت محلية تصل إلى 72% في أي وقت.

هذا ويظهر الجدول F.3 معدل الخطورة لكل خاصية. وفي تقديرات معدلات الخطورة الثلاثة، كانت قيم p كبيرة للغاية بحيث تعذر القيام بأي استنتاجات ملموسة أو الإشارة إلى أن هذه النتائج مهمة من الناحية الإحصائية. وبشكل مناسب، تشمل الفواصل الزمنية الكبيرة للثقة المحيطة بكل تقدير القيمة الافتراضية الباطلة لكل منها، التي تبلغنا، تمامًا كقيم p، أنه ليس بإمكاننا إلغاء القيمة الافتراضية الباطلة عند 1 في أي من تقديرات معدلات الخطورة.

[2] تشير قيمة p منخفضة (أي 0.1 أو أدنى) إلى أهمية النتائج من الناحية الإحصائية. ونظرًا إلى قيم p المرتفعة، لم نتمكن من استنتاج أي ترابطات قيّمة بين خصائص برمجيات إكسبلويت وخطر الموت.

الجدول F.3
تقديرات معدل الخطورة من نماذج كوكس للخطورة النسبية، مع تعديل للخصائص أو بدونه كما هو مبين

خصائص برمجيات إكسبلويت	معدل الخطورة غير المعدل (نماذج منفصلة)	القيمة p	معدل الخطورة المعدل بالكامل	القيمة p	معدل الخطورة المعدل جزئيًا	القيمة p
نوع الثغرة		0.646		0.518		0.458
منطقية	1.17 (0.50, 2.75)	0.715	0.91 (0.36, 2.34)	0.846	0.87 (0.36, 2.11)	0.757
فساد الذاكرة	1.27 (0.59, 2.74)	0.549	0.84 (0.34, 2.05)	0.697	0.83 (0.35, 1.98)	0.680
سوء إدارة الذاكرة	1 (مرجعي)	-	1 (مرجعي)	-	1 (مرجعي)	-
مختلطة أو غير ذلك	2.12 (0.69, 6.53)	0.191	2.04 (0.59, 7.09)	0.264	1.88 (0.61, 5.84)	0.273
منصة متضررة		0.991		0.688		
لينكس	0.78 (0.41, 1.48)	0.447	0.48 (0.18, 1.30)	0.149		
مختلطة	0.89 (0.12, 6.58)	0.905	1.44 (0.18, 11.31)	0.728		
ماك أو.إس	1.26 (0.44, 3.65)	0.668	1.30 (0.41, 4.12)	0.652		
مفتوحة المصدر	0.92 (0.40, 2.14)	0.845	0.53 (0.18, 1.59)	0.259		
غير ذلك	1.15 (0.27, 4.91)	0.853	0.60 (0.11, 3.17)	0.549		
بي إتش بي	1.09 (0.33, 3.62)	0.892	1.60 (0.24, 10.72)	0.626		
قائمة على يونكس	1.07 (0.41, 2.83)	0.887	0.61 (0.19, 1.97)	0.412		
ويندوز	1 (مرجعي)	-	1 (مرجعي)	-		
نوع المصدر[a]		0.732		0.148		0.469
مفتوح	1.09 (0.66, 1.82)	0.732	1.93 (0.79, 4.67)	0.148	1.24 (0.70, 2.19)	0.469
مغلق	1 (مرجعي)	-	1 (مرجعي)	-	1 (مرجعي)	-
نوع الفئة		0.224		0.224		0.239
من جانب العميل	1.02 (0.46, 2.27)	0.969	0.93 (0.40, 2.20)	0.872	1.04 (0.46, 2.37)	0.925
محلية	0.72 (0.41, 1.27)	0.255	0.62 (0.30, 1.32)	0.214	0.60 (0.31, 1.17)	0.134
مختلطة	0.64 (0.25, 1.69)	0.373	0.65 (0.24, 1.78)	0.406	0.61 (0.23, 1.62)	0.318
غير ذلك	0.26 (0.06, 1.12)	0.071	0.15 (0.03, 0.83)	0.030	0.23 (0.05, 1.04)	0.057
عن بعد	1 (مرجعي)	-	1 (مرجعي)	-	1 (مرجعي)	-

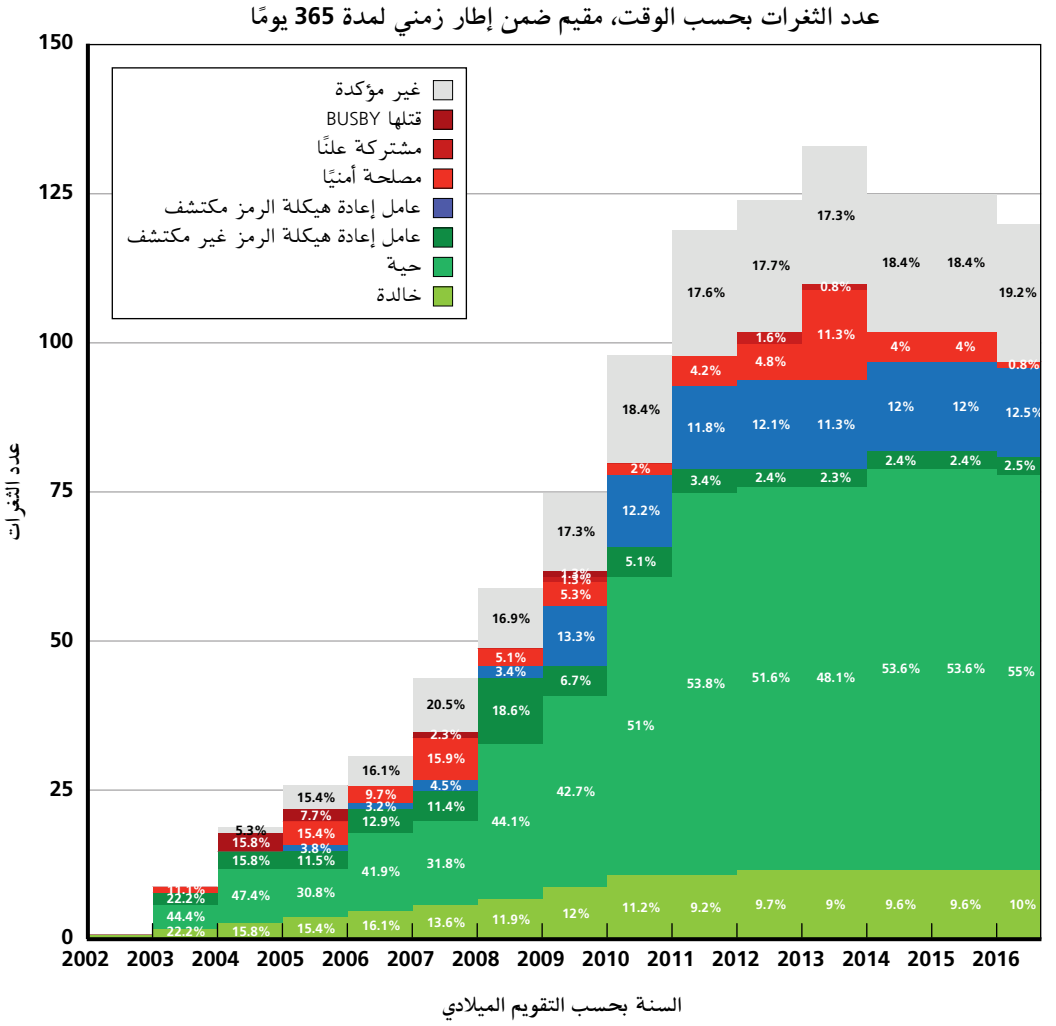
[a] تمّ استثناء برمجية إكسبلويت واحدة تتمتع بنوع مصدر مختلط وست برمجيات مجهولة المصدر من هذا التحليل.

الجدول F.4
أنواع الثغرات بحسب الفاصل الزمني، النسب (n=192)

الحالة	فاصل زمني لمدة 365 يومًا (متوسط، انحراف معياري)	فاصل زمني لمدة 90 يومًا (متوسط، انحراف معياري)	فاصل زمني لمدة 30 يومًا (متوسط، انحراف معياري)
المجموعات الموسعة			
خالدة	11.54 (22.2)	12.79 (26.3)	12.79 (26.79)
حية	47.745 (13.98)	49.38 (16.48)	48.84 (16.56)
عامل إعادة هيكلة الرمز غير مكتشف	4.23 (6.85)	4.9 (8.19)	4.72 (8.66)
عامل إعادة هيكلة الرمز مكتشف	11.52 (5.33)	10.43 (5.54)	11.01 (5.63)
مصلحة أمنيًا	4.52 (5.28)	0.85 (3.3)	0 (2.02)
مكتشفة علنًا	0 (0.52)	0 (0.26)	0 (0.16)
قتلها BUSBY	0 (4.24)	0 (3.59	0 (2.05)
مجهولة	17.49 (6.71)	18.6 (7.47)	18.8 (7.62)
المجموعات المتفككة			
حية	66.86 (11.48)	69.11 (11.37)	69.45 (11.87)
ميتة في هذا الإطار الزمني	5.765 (6.79)	0.87 (5.3)	0 (3.05)
عامل إعادة هيكلة الرمز غير مكتشف	11.52 (5.33)	10.43 (5.54)	11.01 (5.63)
مجهولة	17.49 (6.71)	18.6 (7.47)	18.8 (7.62)

الشكل F.3

فاصل زمني لمدة 365 يومًا، مع النسب المئوية بحسب المجموعة والفاصل الزمني



RAND *RR1751-F.3*

الملحق G

المزيد من المعلومات حول البيانات

تغطي بياناتنا العديد من الفئات. في ما يلي تعداد التواتر في بياناتنا لتوفير حس من الغنى ومعلومات موسعة.

وبسبب الحساسيات، لم ندرج جدولًا بجميع البائعين المتضررين بل نقدم عوضًا عن ذلك الملخص التالي: تشمل هذه المجموعة من البيانات برمجيات إكسبلويت لـ64 بائعًا، مع نطاق يتراوح بين 1 و55 برمجية إكسبلويت للبائع. تمتع كل من مايكروسوفت (n=55) ولينكس (n=39) وآبل (n=14) وصن/أوراكل (n=11) بالعدد الأكبر من برمجيات إكسبلويت، في حين كانت لدى غالبية البائعين الآخرين برمجية إكسبلويت واحدة أو اثنتين. وتشمل هذه اللائحة بائعين ذائعي الصيت على غرار موزيلا (Mozilla) ولينكسيس (LinkSys) وغوغل (Google) وسيتريكس (Citrix) وإيه أو أل (AOL) وإيثريل (Ethereal) وأدوبي (Adobe) وألت–أن تكنولوجيز (Alt-N Technologies) وكريبتوكات (CryptoCat) وريل بلاير/ريل سيرفر (RealPlayer/RealServer)، إلى جانب بعض البائعين الأقل شهرةً.

الجدول G.1
تعداد تواتر البيانات

عناوين السطر	التعداد
الحالة	
حية	66
عامل إعادة هيكلة الرمز	21
خالدة	13
قتلها BUSBY	8
غير ذلك – ميزة غير قابلة للإصلاح أو لم تعد مجدية أو تمت إعادة موضعتها في جزء آخر من الرمز	3
إفصاح علني/معلنة	6
إصلاح أمني	69
التحقق من الفريق	16
مجهولة	5

الجدول G.1 - يتبع

عناوين السطر	التعداد
المجموع	207
نوع الثغرة (مستوى مرتفع)	
سمات مخفية	1
منطقية	57
منطقية، مشفرة	1
منطقية، سوء إدارة الذاكرة	7
منطقية، سوء إدارة الذاكرة، فساد الذاكرة	2
فساد الذاكرة	106
فساد الذاكرة، سوء إدارة الذاكرة	2
سوء إدارة الذاكرة	29
سوء إدارة الذاكرة، قطع الخدمة	1
غير محددة (معطلة)	1
المجموع	207
نوع الثغرة (مستوى منخفض)	
تسمح بتوزيع الذاكرة بطريقة لا يجدر استخدامها	1
سوء استخدام واجهة برمجة التطبيقات	3
حرة عشوائيًا	1
سياق ترابطي	1
تجنب التصديق	5
التنفيذ الذاتي	1
تدفق مفرط للصناديق التي تبدأ برموز (BSS Overflow) (على غرار التدفقات المفرطة المكونة)	1
التجنب	1
سوء إدارة بوابة الاتصالات – عدم التحقق من الأطر	2
ضخ الأوامر	3
تبديل السياق، عدم تماشي التكوين	1
تدفق البيانات المفرط	1
سوء استخدام التصميم	1
اجتياز التوجيهات، تصديق المدخلات	1
هجوم نظام أسماء النطاقات	1
إدخال عامل البيئة (يمكن إدخال برامج أخرى للتنفيذ)	1
تحميل ملف قابل للتنفيذ	1
إصلاح الأخطاء (وحدة المعالجة المركزية)	1
خطأ تطبيع الملف (تفسير مختلف القيم في منصات مختلفة)	1

الجدول G.1 - يتبع

عناوين السطر	التعداد
ميزة قراءة الملف (أي يمكن قراءة الملف في النظام)	1
ميزة قراءة الملف (أي يمكن قراءة الملف في النظام)، حقن إس كيو إل (SQLi)، حقن المسار	1
تدفقات مفرطة مكدسة	58
تدفقات مفرطة مكدسة (نتيجة التدفق المفرط للأعداد الصحيحة)	1
مشكلة دمج مراقب الأجهزة الافتراضية	1
الإفصاح عن المعلومات، سوء إدارة النواة	1
تسريب المعلومات	3
تسريب المعلومات، تدفقات مفرطة مكدسة	1
سوء إدارة الأعداد الصحيحة	1
تدفقات الأعداد الصحيحة المفرطة	2
تقصير الأعداد الصحيحة (للحصول على تدفقات مفرطة مكدسة أو مكونة)	1
إسناد مؤشرى غير صالح	1
مراقبة المدخلات والمخرجات استنادًا إلى مبادئ الكتابة	1
تكامل التواصل الداخلي بين العمليات	1
خفض مجال المفاتيح	1
يؤدي إلى تسريب المعلومات	1
كلمة مرور المدير الضمنية المتبقية	1
منطقي (حالة السباق) يؤدي إلى فساد الذاكرة	1
تصديق الاسم	1
إسناد مؤشري باطل	12
ضخ/إلغاء تسلسل الغرض	4
الصياغة خارج الحدود	1
الأذونات ضمن جهاز نواتي	1
تصعيد الميزات	2
قراءة ملف الميزات	1
سوء إدارة الميزات	2
الهجوم الأمني على الميزات	1
حالة سباق	16
حالة سباق تؤدي إلى إسناد مؤشري غير صالح	1
حالة سباق، تسمح بالتحكم بالذاكرة	1

الجدول G.1 - يتبع

عناوين السطر	التعداد
حالة سباق، خارج حدود القراءة، التدفقات المفرطة المكدسة	1
حالة سباق، لاستخدامها بعد الفترة المجانية	1
حالة مرجعية	2
حالة مرجعية– تؤدي إلى سوء إدارة الأشياء	1
تغطية الحالة المرجعية	1
سوء إدارة السجل/الذاكرة	1
ضخ الرموز عن بعد	1
حقن إس كيو إل	1
تدفقات مفرطة مكونة	40
تدفقات مفرطة مكونة (x2)، الذاكرة	1
تدفقات مفرطة مكونة؛ تدفقات مفرطة مكدسة	1
تشويش في النوع/سوء إدارة الأشياء	1
متغيرات بيئية غير آمنة (للاستفادة في نهاية المطاف من المتغيرات البيئية المحمية)	1
للاستخدام بعد الفترة المجانية	2
يستخدم قيمة غير محققة تدل على الإمدادات (يتبع مؤشرًا عشوائيًا	2
البرمجة عبر المواقع	1
المجموع	207
فئة برمجية إكسبلويت	
التجنب	1
من جانب العميل	25
من جانب العميل، عن بعد	3
من جانب العميل، عن بعد، قطع الخدمة	1
تعتمد على سياق	8
التشفير	1
قطع الخدمة	3
تسريب المعلومات	3
محلية	76
محلية، عن بعد	5
محلية، تسريبات في آلية تحديد الوصول	1
الشبكة	1
معدات نفاذ مادية	1
عن بعد	71

الجدول G.1 - يتبع

عناوين السطر	التعداد
تسريبات في آلية تحديد الوصول/مراقب الأجهزة الافتراضية	6
حقن إس كيو إل	1
المجموع	207
المنصة المتضررة	
أندرويد	3
مضمون	1
فري بي إس دي (FreeBSD)	3
جافا (Java)	1
جافا سكريبت (JavaScript)	1
لينكس	56
لا يوجد	3
مفتوحة المصدر	16
ماك أو إس	18
بي أتش بي	11
سولاريس (Solaris)	9
يونيكس (UNIX)	2
ويندوز	93
المجموع [a]	217
نوع رمز المصدر	
ما من معلومات متاحة	8
مغلق	123
مختلط	2
مفتوح	74
المجموع	207

[a] تعمل بعض برمجيات إكسبلويت على عدة منصات. فبالنسبة إلى 10 برمجيات إكسبلويت، تضررت منصتين أو أكثر.

الملحق H

المسرد

نقدم في اللائحة الواردة أدناه لائحة بالمصطلحات والتعريفات لتعزيز فهم القارئ لثغرات البرمجيات وبرمجيات إكسبلويت. وقد أدرجنا ضمن هذه اللائحة أنواع الثغرات وبرمجيات إكسبلويت التي استعملناها لتصنيف برمجيات إكسبلويت في مجموعة بياناتنا، كما هو محدد في الفصل الثالث.

كما ورد في الفصل الثالث، لا يمكن استثناء أنواع الثغرات بشكل متبادل وغالبًا ما تصبح متشابكة، أو حتى تابعة لبعضها البعض، لا سيما عند أخذ سلاسل برمجيات إكسبلويت الكاملة في الحسبان، والتي غالبًا ما تعتمد على مجموعة من العناصر الأساسية للثغرات تغطي عدة تصنيفات. وقد يؤدي هجوم على تطبيق عبر الويب يقوم أساسًا على ثغرة منطقية إلى فساد الذاكرة أو إلى ضخ أشياء في مجال ذاكرة عملية إطار العمل الكامنة. علاوةً على ذلك، يعتبر نوع الثغرة "المنطقي" واسعًا للغاية ويمكن تقسيمه إلى العديد من الفئات الأخرى. وقد يؤدي خلل في المنطق، على غرار حالة السباق، إلى مشاكل على مستوى منخفض، على غرار فساد الذاكرة وسوء إدارة الذاكرة، إنما أيضًا إلى مشاكل على مستوى أعلى، على غرار ضخ استفسارات عن قاعدة البيانات. غير أنه عمومًا، يمكن للمرء أن يميز بين ثغرات ذاكرة منخفضة المستوى تؤثر كثيرًا على العميلة (أي تتسبب بانهيار عملية) وثغرات منطقية عالية المستوى تسمح لمهاجم بالتأثير على مسار البرنامج لكنها لا تهدد بالضرورة توافر العملية الكامنة.

حية (تصنيف ثغرة يوم الصفر)	ثغرة مجهولة للعامة
السوق الأسود	تجارة أو تهريب أدوات القرصنة، وخدمات القرصنة وثمار القرصنة لأغراض خبيثة. تباع الثغرات للاستعمال الإجرامي أو لأغراض غير مشروعة وتبقى خاصة.
تجاوز سعة المخزن المؤقت	حالة تحدث عندما يسعى برنامج إلى وضع كمية بيانات في مخزن مؤقت تتخطى قدرته الاستيعابية أو عندما يحاول برنامج وضع البيانات في مجال الذاكرة بما يتخطى سعة المخزن (مشروع أمن تطبيقات الإنترنت المفتوحة، غير مؤرخ –a)

الخطأ	عيوب في البرمجيات
المكافأة الأمنية	برنامج وضع لتقديم مكافأة لقاء اكتشاف خطأ أو ثغرة أو الإبلاغ عنهما في منتج برمجية معين (تكنوبيديا Techopedia، غير مؤرخ). وقد تشير أيضًا إلى المكافأة بحد ذاتها.
من جانب العميل (نوع فئة برمجية إكسبلويت)	نوع من برمجيات إكسبلويت المحلية تتطلب تفاعلًا بشريًا لإتمام العملية.
مزيج الرموز	رموز يجري تحدثيها (أو إعادة هيكلتها) إلى نسخ جديدة. قد تظل الثغرات موجودة في النسخ القديمة. كما أنه يشبه عامل إعادة هيكلة الرمز.
عامل إعادة هيكلة الرمز (تصنيف ثغرة يوم الصفر)	ثغرة مجهولة للعامة على الأرجح في النسخ السابقة لمنتج لم يعد قابلًا للاستغلال في نسخ حالية بسبب مراجعات الرموز. تستمر صيانة المنتج (لذا يكون الإصلاح الأمني ممكنًا في المستقبل للنسخ القديمة). وتعرف ثغرات يوم الصفر المماثلة أيضًا بأشباه الأحياء.
الاصطدام، معدل الاصطدام	يحدث الاصطدام عندما يكتشف باحثان (أو أكثر) بشكل مستقل الثغرة نفسها. يشكل احتمال حدوث ذلك معدل الاصطدام. ويشار أحيانًا إلى معدل الاصطدام على أنه معدل التداخل.
أنظمة الثغرات الشائعة والانكشاف	قاموس من الأسماء الشائعة (أي محددات أنظمة الثغرات الشائعة والانكشاف) لثغرات الأمن الإلكتروني المعلومة للآخرين (مايتر MITRE، غير مؤرخ)
ميتة (تصنيف ثغرة يوم الصفر)	ثغرة معلومة للآخرين
قطع الخدمة	هجوم يركز على جعل مصدر (موقع أو تطبيق أو خادم) غير متوافر للغرض الذي صمم لأجله (مشروع أمن تطبيقات الإنترنت المفتوحة OWASP، غير مؤرخ–b)
برمجية إكسبلويت	رمز خبيث يستغل ثغرات البرمجيات لإفساد حاسوب أو تعطيله أو السيطرة عليه بدون موافقة المستخدم، وعادةً بدون معرفته (مايكروسوفت، 2013). توفر برمجية إكسبلويت نفاذًا أوليًا وغالبًا القدرة على تنفيذ الرمز من خلال الاستفادة من بعض الثغرات في عملية النظام، ومن ثم تسهل عمل مغروس أو حمولته.
الفحص العشوائي، الفاحص العشوائي	يشير الفحص العشوائي إلى اختبار البرمجيات العشوائي لاكتشاف الثغرات عادةً. فالفاحص العشوائي هو برنامج يجري مثل هذا الاختبار.

السوق الحكومي	راجع السوق الرمادي
السوق الرمادي	سوق ثغرات تبقى فيه هذه الأخيرة خاصة وتستخدم إما لأغراض الهجوم أو الدفاع وقد تكشف في نهاية المطاف إلى البائع المتضرر، رغم أن الأمر ليس مضمونًا لأنها تباع أولًا إلى حكومة أو جيش أو مقاول يعمل في مجال الدفاع. ويشار إلى السوق الرمادي أيضًا على أنه السوق الحكومي.
التكديس	مجال ضمن مخزن الحاسوب الرئيسي (الذاكرة) محجوز مسبقًا يمكن لعملية برنامج استخدامه لتخزين البيانات بمقدار متغير سيبقى مجهولًا إلى حين تشغيل البرنامج (تيكتارغت Techtar-get، لا تاريخ–a)
خالدة (تصنيف ثغرة يوم الصفر)	ثغرة مجهولة للعامة لنسخة المنتج الذي استحدثت لأجله. تم توقيف صيانة المنتج (لذا لا يمكن إصدار إصلاح أمني أبدًا).
المغروس	برنامج يعزز النفاذ الذي وفرته برمجية إكسبلويت في الدرجة الأولى ويحفظه (أي يحقق الاستمرارية) ويحدث أثرًا معينًا في النظام.
النواة	برنامج يشكل نواة نظام تشغيلي في الحاسوب. يسيطر بالكامل على كل ما يجري في النظام (مشروع معلومات لينكس، غير مؤرخ)
حية (تصنيف ثغرة يوم الصفر)	ثغرة مجهول للعامة للنسخ الحالية من المنتج، لم يكتشفها ويذكرها أحد علنًا (إلى المدى المعلوم)، ويكون الذين يضطلعون بأدوار الدفاع بصدد البحث عنها.
محلية (نوع فئة برمجية إكسبلويت)	برمجية إكسبلويت تتطلب نفاذًا مسبقًا إلى نظام هش (أي تصعيد الميزة)
منطقية (نوع الثغرة)	ثغرات تستند أسسها إلى عيب في منطق البرنامج على مستوى أعلى (مثلًا حالات السباق).
فساد الذاكرة (نوع الثغرة)	ثغرات تستند أسسها إلى فساد منخفض المستوى لذاكرة المعالجة (مثلًا تجاوز سعة المخزن المؤقت)
سوء إدارة الذاكرة (نوع الثغرة)	ثغرات تستند أسسها إلى سوء إدارة منخفضة المستوى لذاكرة المعالجة (مثلًا الاستخدام بعد الفترة المجانية)
التداخل، معدل التداخل	عندما يكتشف باحثان (أو أكثر) بشكل مستقل الثغرة نفسها، يقال إن الثغرات متداخلة. ويمثل معدل التداخل، الذي يعرف أيضًا بمعدل الاصطدام، احتمال أن تتضمن ثغرة ما تداخلًا.

اختبار الانتشار	اختبار لتحديد طريقة تفاعل النظام مع هجوم، سواء كان يمكن اختراق دفاعاته أو لا، ما هي المعلومات التي يمكن الحصول عليها من النظام (كروتز Krutz وفاينز Vines، 2006)
العناصر الأساسية	وحدات البناء الأساسية للثغرات التي يمكن أن تستخدم غالبًا في برمجيات إكسبلويت متعددة.
إثبات المفهوم	إثبات إمكانية تطوير برمجية إكسبلويت فعالة بالكامل في نظام مستهدف. غير أن إثبات مفهوم لا يشمل الخطوات النهائية لتسليح برمجية إكسبلويت (يقوم بذلك عملاء يتمتعون بخصائص معينة حيال البيئة المستهدفة وتشمل قدرات التشويش أو التهرب الضرورية). إن اختبار إثبات المفهوم هو القدرة على جعل برنامج الآلة الحاسبة (calc.exe) يفتح على سطح المكتب (اختبار "بروز الآلة الحاسبة").
عن بعد (نوع فئة برمجية إكسبلويت)	برمجية إكسبلويت لا تتطلب نفاذًا مسبقًا إلى نظام هش. قد تسمح برمجية إكسبلويت أو لا لمشغل بالقيام بمهام إضافية، بما أن النظام الذي قد يستهدفه المهاجم قد يتمتع بميزات منخفضة المستوى. وفي معظم الأحيان، تتطلب عملية ناجحة مزيجًا من برمجيات إكسبلويت المحلية ومن جانب العميل وبرمجيات إكسبلويت عن بعد.
تنفيذ الرمز عن بعد	القدرة على إطلاق تنفيذ عشوائي للرمز من آلة إلى أخرى (ويكيبيديا Wikipedia، غير مؤرخ–a).
آلية تحديد الوصول	بيئة مراقبة عن كثب تحدّ الأذونات على ما يمكن تشغيله (من أجل منع الرمز الخبيث من تنفيذ شيء لا يجدر به القيام به أو النفاذ إلى شيء لا يجب ولوجه، على سبيل المثال).
تسريبات في آلية تحديد الوصول	رمز قادر على تجنب آلية تحديد الوصول
التكوين	منطقة بيانات أو مخزن يستخدم لتخزين الطلبات التي يجب معالجتها (تيكتارغت، غير مؤرخ–b)
الثغرة	برمجية أو معدات أو إجراءات أو ضعف بشري قد يتيح أمام المهاجم مجالًا يمكن استخدامه للاستغلال. نوع من الأخطاء يؤدي إلى ضعف أمني في تصميم نظام أو تطبيقه أو تشغيله (المجلس الوطني للأبحاث NRC، 1999).

السوق الأبيض	سوق للثغرات تعاد فيه الثغرات فورًا إلى البائع المتضرر (غالبًا ما تصبح معلومة للعامة) لأغراض دفاعية. وتشمل برامج المكافأة الأمنية والبائعين ومؤشرات الثغرات ومجموعات على غرار المشروع صفر في غوغل.
ثغرات يوم الصفر	ثغرات قابلة للاستغلال يجهل بائع البرمجيات وجودها ولم يتم إصدار أي إصلاح لها علنًا.
أشباه الأحياء	انظر عامل إعادة هيكلة الرمز

المراجع

Ablon, Lillian, Martin C. Libicki, and Andrea A. Golay, *Markets for Cybercrime Tools and Stolen Data: Hackers' Bazaar,* Santa Monica, Calif.: RAND Corporation, RR-610-JNI, 2014. As of January 30, 2017:
http://www.rand.org/pubs/research_reports/RR610.html

Alberts, Bas, and Massimiliano Oldani, "Beating up on Android—Practical Android Attacks," Immunity Inc., no date. As of 2011:
http://archives.scovetta.com/pub/conferences/infiltrate_2011/Android_Attacks.odt.pdf

Allodi, Luca, and Fabio Massacci, "Comparing Vulnerability Severity and Exploits Using Case-Control Studies," ACM Transactions on Information and System Security (TISSEC), Vol. 17, No. 1, Article 1, August 2014. As of January 30, 2017:
http://disi.unitn.it/~allodi/allodi-tissec-14.pdf

Altinkemer, Kemal, Jackie Rees, and Sanjay Sridhar, "Vulnerabilities and Patches of Open Source Software: An Empirical Study," *Journal of Information System Security*, Vol. 4, No. 2, 2008, pp. 3–25.

Asghari, Hadi, Michael Ciere, and Michel J. G. van Eeten, "Post-Mortem of a Zombie: Conficker Cleanup After Six Years," paper presented at the 24th USENIX Security Symposium, Washington, D.C., August 12–14, 2015. As of January 30, 2017:
https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-asghari.pdf

Bilge, Leyla, and Tudor Dumitras, "Before We Knew It: An Empirical Study of Zero-Day Attacks in the Real World," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, New York: ACM, 2012. As of January 30, 2017:
https://users.ece.cmu.edu/~tdumitra/public_documents/bilge12_zero_day.pdf

Bland, J. Martin, "Survival Probabilities (the Kaplan-Meier Method)," *BMJ*, December 5, 1998.

Brouwer, Andries, "Hackers Hut," 2013. As of January 30, 2017:
https://www.win.tue.nl/~aeb/linux/hh/hh.html#toc9

BugCrowd, "What's a Bug Worth?" 2015. As of January 30, 2017:
https://pages.bugcrowd.com/whats-a-bug-worth-2015

———, "The State of Bug Bounty," June 2016. As of January 30, 2017:
https://pages.bugcrowd.com/hubfs/PDFs/state-of-bug-bounty-2016.pdf

Clark, Sandy, Stefan Frei, Matt Blaze, and Jonathan Smith, "Familiarity Breeds Contempt: The Honeymoon Effect and the Role of Legacy Code in Zero-Day Vulnerabilities," in *Proceedings of the 26th Annual Computer Security Applications Conference*, New York: ACM, 2010, pp. 251–260.

Clarke, Richard A., Michael J. Morell, Geoffrey R. Stone, Cass R. Sunstein, and Peter Swire, *Liberty and Security in a Changing World: Report and Recommendations of the President's Review Group on Intelligence and Communications Technologies*, December 12, 2013. As of February 6, 2017: https://www.justsecurity.org/wp-content/uploads/2013/12/2013-12-12_rg_final_report.pdf

Crocker, Andrew, "What to Do About Lawless Government Hacking and the Weakening of Digital Security," Electronic Frontier Foundation, August 1, 2016. As of January 30, 2017: https://www.eff.org/deeplinks/2016/08/what-do-about-lawless-government-hacking-and-weakening-digital-security

Dai Zovi, Dino, Twitter post, August 1, 2016, https://twitter.com/dinodaizovi/status/760744230471540742

Daniel, Michael, "Heartbleed: Understanding When We Disclose Cyber Vulnerabilities," White House archives, April 28, 2014. As of January 30, 2017: https://obamawhitehouse.archives.gov/blog/2014/04/28/heartbleed-understanding-when-we-disclose-cyber-vulnerabilities

Defense Advanced Research Projects Agency, Cyber Grand Challenge Competitor Portal, website, 2016: As of January 30, 2017: https://cgc.darpa.mil/

Exodus Intelligence, "Changing to Coordinated Disclosure," May 17, 2016. As of January 30, 2017: https://blog.exodusintel.com/2016/02/18/changing-to-coordinated-disclosure/

Ferdinando, Lisa, "Carter Announces 'Hack the Pentagon' Program Results," U.S. Department of Defense, June 17, 2016. As of January 30, 2017: http://www.defense.gov/News/Article/Article/802828/carter-announces-hack-the-pentagon-program-results

Fidler, Mailyn, Jennifer Granick, and Martha Crenshaw, *Anarchy or Regulation: Controlling the Global Trade in Zero-Day Vulnerabilities*, master's thesis, Stanford University, 2014. As of January 30, 2017: https://stacks.stanford.edu/file/druid:zs241cm7504/Zero-Day%20Vulnerability%20Thesis%20by%20Fidler.pdf

Finifter, Matthew, Devdatta Akhawe, and David Wagner, "An Empirical Study of Vulnerability Rewards Programs." *USENIX Security*, Vol. 13, 2013. As of January 30, 2017: https://www.cs.berkeley.edu/~daw/papers/vrp-use13.pdf

Finkle, Jim, "Apple Offers Big Cash Rewards for Help Finding Security Bugs," Reuters, August 5, 2016. As of January 30, 2017: http://www.reuters.com/article/us-cyber-blackhat-apple-idUSKCN10F2TX

Geer, Dan, "Cybersecurity as Realpolitik," BlackHat, August 6, 2014. As of January 30, 2017: http://geer.tinho.net/geer.blackhat.6viii14.txt

Gellman, Barton, and Ellen Nakashima, "U.S. Spy Agencies Mounted 231 Offensive Cyber-Operations in 2011, Documents Show," *Washington Post*, August 30, 2013. As of January 30, 2017: https://www.washingtonpost.com/world/national-security/us-spy-agencies-mounted-231-offensive-cyber-operations-in-2011-documents-show/2013/08/30/d090a6ae-119e-11e3-b4cb-fd7ce041d814_story.html

Goel, Manish Kumar, Pardeep Khanna, and Jugal Kishore, "Understanding Survival Analysis: Kaplan-Meier Estimate," International Journal of Ayurveda Research, Vol. 1., No. 4, October–December 2010, pp. 274–278.

Greenberg, Andy, "Shopping for Zero-Days: A Price List for Hackers' Secret Software Exploits," Forbes.com, March 23, 2012. As of January 30, 2017:
http://www.forbes.com/sites/andygreenberg/2012/03/23/shopping-for-zero-days-an-price-list-for-hackers-secret-software-exploits/

———, "Here's a Spy Firm's Price List for Secret Hacker Techniques," *Wired*, November 11, 2015. As of January 30, 2017:
https://www.wired.com/2015/11/heres-a-spy-firms-price-list-for-secret-hacker-techniques/

HackerOne, "How Should We Price Bounties?" 2016. As of September 14, 2016:
https://support.hackerone.com/hc/en-us/articles/204950689-How-should-we-price-bounties-

Hadzima, Joe, "How Much Does an Employee Cost?" *Boston Business Journal*, 2005. As of January 30, 2017:
http://web.mit.edu/e-club/hadzima/how-much-does-an-employee-cost.html

hdmoore, "Security Flaws in Universal Plug and Play: Unplug, Don't Play," Rapid7Community, January 29, 2013. As of January 30, 2017:
https://community.rapid7.com/community/infosec/blog/2013/01/29/security-flaws-in-universal-plug-and-play-unplug-dont-play

Hosenball, Mark, "FBI Paid Under $1 Million to Unlock San Bernardino iPhone: Sources," Reuters, May 4, 2016. As of January 30, 2017:
http://www.reuters.com/article/us-apple-encryption-idUSKCN0XQ032

Jurczyk, Mateuz, "Windows Kernel Trap Handler and NTVDM Vulnerabilities—Case Study," 2013. As of September 14, 2016:
http://www.slideshare.net/DefconRussia/mateusz-j00ru-jurczyk-windows-kernel-trap-handler-and-ntvdm-vulnerabilities-case-study

Kharpal, Arjun, "Ethical Hacking: Are Companies Ready?" CNBC, June 17, 2015. As of January 30, 2017:
http://www.cnbc.com/2015/06/17/are-companies-still-scared-of-white-hat-hackers.html

Kirda, Engin, Andrea Mambretti, Wil Robertson, Brendan Dolan-Gavitt, Patrick Hulin, and Frederich Ulrich, "LAVA: Large-scale Automated Vulnerability Addition," presentation, New York University, no date. As of January 30, 2017:
http://panda.moyix.net/~moyix/LAVA_Sapienza.pdf

Krutz, Ronald L., and Russell Dean Vines, *The CISSP® and CAPCM Prep Guide: Platinum Edition*, New York: John Wiley & Sons, 2006.

Kuehn, Andreas, and Milton Mueller, "Analyzing Bug Bounty Programs: An Institutional Perspective on the Economics of Software Vulnerabilities," paper presented at 2014 TPRC: The 42nd Research Conference on Communication, Information and Internet Policy, August 1, 2014.

Lam, Mimi, presentation explaining the purpose of the Kaplan-Meier estimator, Vanderbilt University, Department of Biostatistics wiki, no date. As of January 30, 2017:
http://biostat.mc.vanderbilt.edu/wiki/pub/Main/ClinStat/km.lam.pdf

Libicki, Martin C., Lillian Ablon, and Tim Webb, *Defender's Dilemma: Charting a Course Toward Cybersecurity*, Santa Monica, Calif.: RAND Corporation, RR-1024-JNI, 2015. As of January 30, 2017:
http://www.rand.org/pubs/research_reports/RR1024.html

linfo, "Kernel Definition," undated. As of January 31, 2017:
http://www.linfo.org/kernel.html

Maillart, Thomas, Mingyi Zhao, Jens Grossklags, and John Chuang, "Given Enough Eyeballs, All Bugs Are Shallow? Revisiting Eric Raymond with Bug Bounty Program," University of California, Berkeley, 2016. As of January 30, 2017:
http://weis2016.econinfosec.org/wp-content/uploads/sites/2/2015/08/WEIS_2016_paper_76.pdf

McConnell, Steve, *Code Complete: A Practical Handbook of Software Construction*, 2nd edition, Redmond, Wash.: Microsoft Press, 2004.

Microsoft, "Security Development Lifecycle: Life in the Digital Crosshairs," no date. As of January 30, 2017:
https://www.microsoft.com/security/sdl/story/

———, *Microsoft Security Bulletin*, 1997–2016.

———, *Microsoft Security Intelligence Report (MSIR)*, Vol. 15, January–June 2013. As of January 30, 2017:
http://download.microsoft.com/download/5/0/3/50310CCE-8AF5-4FB4-83E2-03F1DA92F33C/Microsoft_Security_Intelligence_Report_Volume_15_English.pdf

Microsoft Tech Center, "Bounty Hunters: The Honor Roll," 2016a. As of January 30, 2017:
https://technet.microsoft.com/en-us/security/dn469163

———, "Microsoft Bounty Programs," 2016b. As of January 30, 2017:
https://technet.microsoft.com/en-us/security/dn425036

Miller, Charlie, *The Legitimate Vulnerability Market: Inside the Secretive World of 0-Day Exploit Sales*, Independent Security Evaluators, May 6, 2007. As of January 30, 2017:
http://weis2007.econinfosec.org/papers/29.pdf

MITRE, "Common Vulnerabilities and Exposures," undated. As of January 31, 2017:
https://cve.mitre.org/about/

Moore, Tyler, Allan Friedman, and Ariel D. Procaccia, "Would a 'Cyber Warrior' Protect Us: Exploring Trade-Offs Between Attack and Defense of Information Systems," in *Proceedings of the 2010 Workshop on New Security Paradigms*, New York: ACM, 2010, pp. 85–94.

Moussouris, Katie, and Michael Siegel, *The Wolves of Vuln Street: The 1st System Dynamics Model of the 0day Market*, presentation given at RSA Conference 2015, San Francisco, April 20–24, 2015. As of January 30, 2017:
https://www.rsaconference.com/writable/presentations/file_upload/ht-t08-the-wolves-of-vuln-street-the-1st-dynamic-systems-model-of-the-0day-market_final.pdf

Naraine, Ryan, "Stuxnet Attackers Used 4 Windows Zero-Day Exploits," ZDNet, September 14, 2010. As of January 30, 2017:
http://www.zdnet.com/article/stuxnet-attackers-used-4-windows-zero-day-exploits/

National Research Council, *Trust in Cyberspace*, Fred B. Schneider, ed., Washington, D.C.: The National Academies Press, 1999.

———, *Technology, Policy, Law, and Ethics Regarding U.S. Acquisition and Use of Cyberattack Capabilities*, William A. Owens, Kenneth W. Dam, Herbert S. Lin, eds., Washington, D.C.: The National Academies Press, 2009.

National Vulnerability Database, no date. As of February 1, 2017:
http://nvd.nist.gov/

Newsome, James, and Dawn Song, "Dynamic Taint Analysis for Automatic Detection, Analysis, and Signature Generation of Exploits on Commodity Software," Carnegie Mellon University, School of Computer Science, July 2005. As of January 30, 2017:
http://bitblaze.cs.berkeley.edu/papers/taintcheck-full.pdf

No Starch Press, "Security," web page, 2016. As of January 30, 2017:
https://www.nostarch.com/catalog/security

NRC—*See* National Research Council.

Open Web Application Security Project, "Buffer Overflow," undated-a. As of January 21, 2017:
https://www.owasp.org/index.php/Buffer_Overflow

———, "Denial of Service," undated-b. As of January 31, 2017:
https://www.owasp.org/index.php/Denial_of_Service

OWASP—*See* Open Web Application Security Project.

Ozment, Andy, and Stuart E. Schechter, "Milk or Wine: Does Software Security Improve with Age?" paper presented at the 15th USENIX Security Symposium, August 3, 2006. As of January 30, 2017:
https://www.usenix.org/legacy/event/sec06/tech/ozment.html

Pagliery, Jose, "You Make $70k but Cost Your Boss $88k," CNN Money, February 28, 2013. As of January 30, 2017:
http://money.cnn.com/2013/02/28/smallbusiness/salary-benefits/

Ransbotham, S., and S. Mitra, "The Impact of Immediate Disclosure on Attack Diffusion and Volume," in 10th Workshop on Economics of Information Security (WEIS 2011), Fairfax, Va., June 14–15, 2011.

Rescorla, Eric, "Is Finding Security Holes a Good Idea?" *IEEE Security & Privacy*, Vol. 3, No. 1, January–February 2005, pp. 14–19.

Risk-Based Security, "VulnDB—Vulnerability Intelligence," 2017. As of January 30, 2017:
https://www.riskbasedsecurity.com/vulndb/

Salary.com, "Benefits Wizard," 2016, As of January 30, 2017:
http://swz.salary.com/MyBenefits/LayoutScripts/Mbfl_Start.aspx

Schneier, Bruce, "Should U.S. Hackers Fix Cybersecurity Holes or Exploit Them?" *The Atlantic*, May 19, 2014. As of January 30, 2017:
http://www.theatlantic.com/technology/archive/2014/05/should-hackers-fix-cybersecurity-holes-or-exploit-them/371197/

Schryen, Guido, and Eliot Rich, "Increasing Software Security Through Open Source or Closed Source Development? Empirics Suggest That We Have Asked the Wrong Question," paper presented at the 43rd Hawaii International Conference on System Sciences (HICSS), January 5–8, 2010.

Schwartz, Ari, and Rob Knake, "Government's Role in Vulnerability Disclosure: Creating a Permanent and Accountable Vulnerability Equities Process." Discussion Paper 2016-04, Cyber Security Project, Belfer Center for Science and International Affairs, Harvard Kennedy School, June 2016. As of January 30, 2017:
http://belfercenter.ksg.harvard.edu/files/vulnerability-disclosure-web-final3.pdf

Shieber, Jonathan, "Synack Raises $7.5 Million Putting Bounties on IT Security Threats," *Tech Crunch*, April 14, 2014: As of January 30, 2017
https://techcrunch.com/2014/04/24/synack-raises-7-5-million-putting-bounties-on-it-security-threats/

Shryock, Henry S., Jacob S. Siegel, and Elizabeth A. Larmon, *The Methods and Materials of Demography*, Washington, D.C.: U.S. Bureau of the Census, 1973. As of January 30, 2017: http://demographybook.weebly.com/uploads/2/7/2/5/27251849/david_a._swanson_jacob_s._siegel_the_methods_and_materials_of_demography_second_edition__2004.pdf

StalkR, "Exec Race Condition Exploitations," StalkR's Blog, November 6, 2010. As of January 30, 2017:
http://blog.stalkr.net/2010/11/exec-race-condition-exploitations.html

Techopedia, Bug Bounty, web page, undated-a. As of January 31, 2017:
http://www.techopedia.com/definition/28637/bug-bounty

Techtarget, "heap," undated-a. As of January 31, 2017:
http://whatis.techtarget.com/definition/heap

———, "stack," undated-a. As of January 31, 2017:
http://whatis.techtarget.com/definition/stack

Tsyrklevich, Vlad, "Hacking Team: A Zero-Day Market Case Study," July 22, 2015. As of January 30, 2017:
https://tsyrklevich.net/2015/07/22/hacking-team-0day-market/

U.S. Office of Personnel Management, "Pay & Leave—Pay Administration," 2016. As of January 30, 2017:
https://www.opm.gov/policy-data-oversight/pay-leave/pay-administration/fact-sheets/computing-hourly-rates-of-pay-using-the-2087-hour-divisor/

Wikipedia, "Arbitrary Code Execution," undated-a. As of January 31, 2017:
https://en.wikipedia.org/wiki/Arbitrary_code_execution

Zerodium, "Zerodium Payout Ranges," January 2016. As of January 30, 2017:
https://zerodium.com/images/zerodium_prices.png

Zetter, Kim, "When Security Experts Gather to Talk Consensus, Chaos Ensues," *Wired*, October 1, 2015. As of January 30, 2017:
https://www.wired.com/2015/10/security-experts-gather-talk-consensus-chaos-ensues/

Zhao, Mingyi, Jens Grossklags, and Peng Liu, "An Empirical Study of Web Vulnerability Discovery Ecosystems," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, New York: ACM, 2015.

Zorabedian, John, "How a Hacking Team Got Hacked," *Naked Security*, April 2016. As of January 30, 2017:
https://nakedsecurity.sophos.com/2016/04/19/how-hacking-team-got-hacked/

تعتبر ثغرات يوم الصفر، وهي ثغرات البرمجيات التي لم يتمّ إصدار إصلاحات لها علنًا، وبرمجيات إكسبلويت مفيدة في العمليات الإلكترونية، سواء تلك التي يشنها المجرمون أو الجيوش أو الحكومات، وكذلك في المجالات الدفاعية والأكاديمية.

يوفر هذا التقرير استنتاجات من بيانات ثغرات يوم الصفر وبرمجيات إكسبلويت حقيقية قد تعزز الأمثلة البديلة التقليدية وآراء الخبراء، وتكمل الجهود الحالية لاستحداث إطار عمل لاتخاذ قرار حول الكشف عن مجموعة ثغرات يوم الصفر وبرمجيات إكسبلويت أو الاحتفاظ بها، وتقدم المعلومات اللازمة للجدالات القائمة حول السياسة في ما يخص تخزين الثغرات والإفصاح عنها، وتضيف سياقًا للذين يراقبون التداعيات والمسؤولية المترتبة عن الهجوم وعمليات انتهاك البيانات بالنسبة إلى العملاء والشركات والضامنين والنظام المدني القضائي عمومًا في الولايات المتحدة.

هذا وقد قدم المؤلفون رؤى معمقة حول قطاع البحث عن ثغرات يوم الصفر وتطوير برمجيات إكسبلويت، ووفروا معلومات عن نسبة ثغرات يوم الصفر الحية (غير المفصح عنها) أو الميتة (المعلومة) أو في حالة ما بينهما، وحددوا بعض المقاييس الأساسية المتعلقة بمتوسط أمد حياة ثغرات يوم الصفر، واحتمال إقدام فريق آخر على كشف ثغرة ضمن فترة زمنية محددة، إلى جانب الوقت والتكاليف لتطوير برمجية إكسبلويت لاستغلال ثغرة يوم صفر.

www.rand.org

Arabic translation: *Zero Days, Thousands of Nights: The Life and Times of Zero-Day Vulnerabilities and Their Exploits*
RR-1751/1-RC