Prevenção de ataques *zero-day* por meio de vulnerabilidades divulgadas

Alexandre Y. Kajihara[1]

[1] DACOM - Departamento Acadêmico de Computação do Curso de Bacharelado em Ciência da Computação – Universidade Tecnológica Federal do Paraná (UTFPR)
Campo Mourão – PR – Brazil

alexandre.ykz@gmail.com

Resumo. *Os crimes cibernéticos são delitos realizados por meio de computadores ou da Internet. Um tipo de crime que está aumentando nos últimos tempos é o ataque de dia zero, que explora vulnerabilidades ainda não divulgadas publicamente. Considerando que esses ataques sempre causam grandes prejuízos, o trabalho aqui proposto tem como objetivo auxiliar usuários a prevenir que suas aplicações fiquem expostas e, consequentemente, que seus dados sigilosos sejam divulgados. Para isso, serão utilizadas informações de falhas divulgadas em bases de dados sobre vulnerabilidades. Serão selecionadas as informações mais relevantes, que poderão ser combinadas. Feitas essas associações, pretende-se testar a aplicação para verificar a existência de vulnerabilidades.*

1. Introdução

O crime cibernético é um delito cometido por meio da Internet ou de uma rede de computadores, que pode ser utilizada: como ferramenta de um crime, quando utilizada para cometê-lo; ou como vítima de um crime, no caso de ser o alvo do delito [Paolo Foti 2015].

A Internet, por ser muito aberta e incontrolável, pode acarretar problemas desagradáveis aos seus usuários, como as ciberinfecções, do tipo *trojans*, *exploits* e ferramentas maliciosas. De acordo com o *Kaspersksy Security Network*, 10% das ameaças atuais são *exploits*. É preciso compreender que o *software* é criado por humanos, que podem cometer erros. Além disso, não existe um método "perfeito" de programação. É por isso que um software pode conter vulnerabilidades, isto é, erros no código de programação por onde cibercriminosos podem controlar um sistema, "desarrumá-lo" etc. É justamente o código que explora vulnerabilidades em programas que é denominado de *exploit* [Kaspersky 2012].

As vulnerabilidades podem ser usadas pelos *exploits*, ou podem permanecer inexploradas, dependendo da popularidade e da funcionalidade de cada programa e, consequentemente, da atenção que despertam nos cibercriminosos. Se uma vulnerabilidade, em um determinado programa, não for descoberta, isso não significa que ela não existe, mas sim que o programa é muito pouco utilizado, e por isso dificilmente alguém encontrará uma vulnerabilidade por engano, ou que o programa é tão insignificante que não vale a pena, para os cibercriminosos, procurar erros nele [Kaspersky 2012].

Em um cenário ideal, uma vulnerabilidade é descoberta por um investigador, que a relata ao desenvolvedor do programa. Este, por sua vez, corrige urgentemente a vulnerabilidade por meio de uma atualização. Nesse momento, o submundo cibernético "reune as peças", cria um *exploit* e tenta atacar os usuários que ainda não instalaram a atualização. Esse cenário ideal, com um intervalo de tempo entre a descoberta da vulnerabilidade e o aparecimento do *exploit*, não é o que acontece frequentemente. Isso porque, o que costuma ocorrer, é que o *exploit* surja simultaneamente com as primeiras notícias sobre a vulnerabilidade, ou, ainda pior, que o *exploit* seja lançado antes de a atualização ficar disponível. Esse *exploit*, para o qual ainda não há uma atualização, é chamado de exploit de dia zero (*zero-day*), porque ele é lançado zero dia depois, ou seja, antes da atualização [Kaspersky 2012].

A vulnerabilidade de dia zero refere-se à lacuna de segurança existente em sites, softwares ou aplicativos, mas que ainda não foi descoberta no momento de seu lançamento. Isso significa que o desenvolvedor tem zero dia para corrigir a vulnerabilidade [Pratap Kumar 2016].

Portanto, em um ataque de *exploit* de dia zero é explorada uma vulnerabilidade que ainda não foi divulgada publicamente. É muito difícil se defender de um ataque de dia zero, porque enquanto a vulnerabilidade permanecer desconhecida, o *software* afetado não pode ser corrigido, e os antivírus não podem detectá-lo. Para os cibercriminosos, vulnerabilidades em softwares populares, como o Microsoft Office e o Adobe Flash, são um passe livre para qualquer alvo que eles queiram atacar, como grandes empresas ou milhões de *desktops* ao redor do mundo [Leyla Bilge 2012].

Na década de 2010, portanto, a situação é muito diferente da observada nos anos de 1980, quando provavelmente as primeiras vulnerabilidades de dia zero ocorreram. Naquela época, a sociedade não percebia a importância que a segurança da informação teria em um futuro próximo [Paolo Foti 2015]. Nesse mundo atual, em que a Internet faz, cada vez mais, parte da vida das pessoas e das empresas, é imprescindível identificar vulnerabilidades, como, por exemplo, de dia zero, para aumentar a segurança aos utilizadores de qualquer serviço *Web*. Essa situação, portanto, motivou-nos a realizar este estudo, que tem como objetivo auxiliar usuários a prevenir que as suas aplicações fiquem expostas e, consequentemente, que seus dados sigilosos sejam divulgados para pessoas não autorizadas. Para isso, iremos utilizar os detalhes de falhas já divulgadas em bases de dados sobre vulnerabilidades, e entre essas informações, as mais relevantes serão empregadas, ou combinações das mesmas. Com as associações feitas, desejamos testar a aplicação para verificar a existência de alguma vulnerabilidade, e indicar ao fornecedor do *software* quais ataques podem ser explorados em sua aplicação. Com essa prática de prevenção, os fornecedores de *software* terão mais uma forma de testar suas atuais e novas aplicações, prevenindo-as de possíveis ataques de dia zero.

O trabalho aqui apresentado está organizado da seguinte forma: na seção 2 é apresentado o referencial teórico do estudo, ou seja, definições e questões relacionadas aos ataques de *exploits* de dia zero; na seção 3 são apresentadas pesquisas que analisam diversos aspectos dos ataques de dia zero; na seção 4 é descrito o método do trabalho que se pretende realizar; e na última seção, os resultados esperados do estudo.

2. Referencial teórico

Os ataques de segurança são uma preocupação que têm aumentado muito, na última década. Em 2003, havia cerca de 6,3 bilhões de pessoas vivendo no planeta e 500 milhões de dispositivos conectados à Internet. Sete anos depois, em 2010, o número de dispositivos conectados à Internet aumentou para 12,5 bilhões, em decorrência do uso dos *smartphones* e *tablets*. Em 2020, estima-se que haverá 50 bilhões de dispositivos conectados à Internet [Evans 2011]. Esses dados dão uma ideia de como os riscos aos ataques de segurança são altos, atualmente, e tenderão a crescer, nos próximos anos. Por exemplo, somente no primeiro trimestre de 2012, foram registrados 7.000 exemplares de ameaças (*malware*) móveis ao Android [McAffe 2012].

Os ataques costumam ocorrer na Internet com variados objetivos e alvos. Todo serviço, computador ou rede acessada via Internet pode se tornar alvo de um ataque, assim como qualquer computador que tenha acesso à Internet pode ser usado em um ataque. Diferentes técnicas podem ser utilizadas nos ataques, como, por exemplo, interceptação de tráfego e negação de serviço [CERT.br 2012].

A "negação de serviço" (*Denial of Service* – DoS) consiste em uma técnica em que o atacante utiliza um computador para tirar de operação um serviço, um computador ou uma rede conectada à Internet. Isso é realizado não para invadir ou coletar informações, mas para que usuários não possam acessá-los, por estarem indisponíveis ou extremamente lentos. Na "interceptação de tráfego" (*sniffing*), dados que trafegam em redes de computadores são inspecionados, por meio de programas denominados *sniffers*, com o objetivo de capturar senhas, números de cartões de crédito ou conteúdos de arquivos confidenciais [CERT.br 2012].

Além desses ataques tradicionais, existem técnicas de ataques direcionados, criados para invadir uma determinada rede, ou roubar informações específicas, sem serem detectados. Um exemplo é o *exploit*, que significa, literalmente, "explorar". Um *exploit* é uma técnica que permite tirar proveito de uma falha existente em um *software*, provocando comportamentos não esperados, para conseguir vantagens, controle do sistema ou negar serviços (DoS). Um *exploit* pode ser, por exemplo, um programa executável, uma mensagem em um determinado protocolo de rede ou uma mensagem escondida em um e-mail. Geralmente, um *exploit* pode ser utilizado para tomar vantagem de uma única vulnerabilidade do *software*. Quando um *exploit* é divulgado, a vulnerabilidade é corrigida por meio de uma correção (*patch*), o que faz com que ele se torne obsoleto para as novas versões desse *software*. É por isso que muitos *hackers* não publicam os seus *exploits*, mantendo-os privados. Esses *exploits* não divulgados publicamente são chamados de dia zero [Pedrosa 2007].

Um ataque de *exploit* de dia zero (*zero-day exploit*) explora uma vulnerabilidade de um software ainda desconhecida por seus desenvolvedores, e por isso correções ainda não estão disponíveis [Serge Egelman 2013]. Dessa forma, o *software* não pode ser corrigido e os antivírus não podem detectá-lo. Um ataque de dia zero dura, em média, 312 dias. Entretanto, existem ataques que permanecem desconhecidos por até dois anos e seis meses [Leyla Bilge 2012].

Grande parte dos sistemas de detecção de intrusões é baseado em assinaturas, e por isso não consegue identificar ataques de dia zero. Isso porque esses sistemas

somente conseguem detectar ataques para os quais já têm assinaturas, ou seja, que estejam programados para reconhecer. Dessa forma, uma rede pode estar sendo atacada, mas talvez seja necessário alguns dias até que uma nova assinatura, que detecte o ataque, seja publicada. O problema é que as empresas não podem ficar esperando pelo desenvolvimento e instalação de assinaturas, pois as ameaças se propagam rapidamente. Por exemplo, em 2003, em apenas 26 dias o *worm* Blaster atacou os micros que utilizavam os sistemas operacionais Windows, que não haviam realizado a correção publicada pela Microsoft. Portanto, manter o sistema sempre atualizado é importante para prevenir ataques [McCarthy 2004].

A combinação de sistemas de detecção de intrusões baseados em assinaturas e em protocolos é a ideal para prevenir ataques. A detecção de anomalias de protocolo é feita no protocolo do aplicativo, concentrando-se na estrutura e conteúdo das comunicações. Os protocolos Telnet, HTTP, RPC e SMTP são alvos de muitos ataques. Erros de programação, como, por exemplo, o estouro do buffer, pode ser utilizado pelos invasores para comprometer ou danificar o sistema [McCarthy 2004].

Os sistemas de detecção de intrusões, sejam de assinaturas ou de protocolo, não são capazes de identificar todos os ataques. Se as empresas tiverem implementado apenas assinaturas IDS por toda a rede, não conseguirão identificar os ataques de dia zero. Considerando que as anomalias de protocolo conseguem detectar ataques desconhecidos, é preciso que as empresas reforcem suas defesas nos gates de sua rede, como, por exemplo, na conexão com a Internet, nas conexões VPN e nas conexões de rede dos clientes. Dessa forma, será possível levantar a primeira linha de defesa nos pontos de entrada, de forma a contribuir para que os ataques sejam detectados o mais cedo possível [McCarthy 2004].

Considerando o número crescente das vulnerabilidades nos softwares, organizações têm elaborado bases que integram e fornecem informações sobre essas falhas. Essas bases contribuem para que desenvolvedores construam *softwares* mais seguros e mantenham os antivírus atualizados [Santos 2017]. Exemplos de base de dados são a *National Vulnerability Database* (NVD), do órgão do governo norte-americano *National Institute of Standards and Technology*, e a *Common Vulnerabilities and Exposures* (CVE), da organização norte-americana sem fins lucrativos MITRE, e a *A-Z Listing of Threats & Risks*, da empresa americana Symantec. A CVE fornece atualizações, em tempo real, sobre ameaças atuais e potencialmente futuras, níveis de gravidade e sugestões de correções a serem utilizadas em atualizações [Bertucci 2017]. A NVD oferece informações de quase todas as vulnerabilidades de softwares divulgadas publicamente, e o registro de informações da severidade e facilidade de explorar essas vulnerabilidades [Last 2016].

3. Trabalhos relatados

O tema ataque de dia zero tem sido pesquisado, atualmente, sob diferentes perspectivas. [Last 2016] realizou um trabalho de busca de modelos de descoberta de vulnerabilidades que possam fornecer previsões para a descoberta de vulnerabilidades de dia zero. A partir de dados do banco de vulnerabilidade *National Vulnerability Database*, o pesquisador propôs três processos distintos de previsão de vulnerabilidades de dia zero, um baseado em regressão e dois em aprendizagem de máquinas, e os resultados iniciais do estudo

foram promissores.

Considerando a grande quantidade de aplicações *Web* lançadas nos últimos tempos, e o aumento de ataques de dia zero contra vulnerabilidades nesses aplicativos, [Pratap Kumar 2016] analisaram técnicas que permitam identificar e eliminar *exploits* em tempo real ou minimizar os danos causados por eles. Na técnica baseada em estatística, qualquer acontecimento fora do normal é bloqueado ou marcado. Na técnica baseada em assinatura, usada em antivírus, a detecção depende da existência de uma assinatura previamente conhecida, e por isso não é eficaz para prevenir ataques de dia zero. A técnica baseada no comportamento objetiva prever o futuro comportamento de um servidor *Web*, para negar quaisquer comportamentos que não sejam esperados. A técnica híbrida combina as técnicas baseadas na estatística e no comportamento, e visa detectar *worms* polimórficos de dia zero.

[Leyla Bilge 2012] propuseram uma técnica para identificar, automaticamente, ataques de dia zero, a partir de dados disponíveis no *World Intelligence Network Environment*. Eles registraram quantos binários benignos e maliciosos foram baixados em 11 milhões de hosts reais em todo o mundo, e identificaram 18 vulnerabilidades exploradas antes da divulgação, sendo que 11 tinham sido empregadas em ataques de dia zero.

[Paolo Foti 2015] analisaram as vulnerabilidades de dia zero no contexto mais amplo do cibercrime e dos mercados econômicos. A partir de entrevistas de especialistas, descreveram diferentes mercados de vulnerabilidades de zero dia, ou seja, mercados brancos, negros e cinzas (governamentais), e discutiram características das fábricas de malware e seus principais clientes. Outro trabalho que discutiu a questão do comércio dos *exploits* de dia zero foi realizado por [Serge Egelman 2013]. Estes pesquisadores alertaram para a necessidade de se discutir questões éticas relacionadas aos mercados relacionados aos ataques de dia zero, visto que mercados que facilitam a venda de detalhes de vulnerabilidades estão ganhando popularidade no mundo comercial.

4. Método

Para a prevenção de ataques dia zero será realizada a análise da combinação de dados de vulnerabilidades já divulgadas, em três bases: o National Vulnerability Database – NVD [NIST 2017], a Common Vulnerabilities and Exposures – CVE [MITRE 2017] e a A-Z Listing of Threats & Risks [Symantec].

A análise dessas três bases de dados sobre vulnerabilidades evidencia algumas peculiaridades das informações fornecidas. A *A-Z Listing of Threats & Risk* apresenta mais detalhes que as demais, sobre os tipos de vulnerabilidades: *worm*, vírus, *spyware*, macro, *trojan* ou *trojan horse* etc. Além disso, é possível: visualizar o sistema afetado; a data de descoberta; o que essa vulnerabilidade realiza, com muitos detalhes; se arquivos são criados por causa dessa falha; o risco de impacto; e algumas descrições de como determinadas vulnerabilidades podem ser reproduzidas.

A base Common Vulnerabilities Exposures (CVE) fornece uma lista de vulnerabilidades mais completa que a National Vulnerability Database (NVD). Os registros da NVD contêm informações sobre: a gravidade e facilidade de exploração de vulnerabilidades divulgadas; a complexidade de acesso, se há ou não necessidade

de autenticação, os tipos de impacto e as vulnerabilidades presentes em softwares, por fornecedores. A base CVE fornece uma breve descrição da vulnerabilidade e algumas referências de como reproduzi-las, ou os arquivos que são criados a partir da vulnerabilidade. Como na base CVE são fornecidas poucas informações sobre algumas vulnerabilidades, é preciso fazer uma pesquisa a partir de seu identificador, para coleta de mais dados. Em alguns casos, a CVE não fornece descrições da vulnerabilidade, visto que o fornecedor ainda não as forneceu, e por isso é importante que os dados dessa base sejam sempre atualizados.

As informações dessas três bases são complementares, e a combinação das mesmas pode ser útil para a prevenção de ataques cibernéticos, como os de dia zero. Por exemplo, reproduzindo os ataques descritos nas bases, podemos verificar quais aplicações apresentam essas falhas, quais fornecedores têm softwares mais vulneráveis. Isso permitirá que o usuário opte por um sistema que não possua tantas vulnerabilidades.

Após a realização de um ataque, pode-se verificar se algum arquivo foi criado, que realize sequestro de dados. Também é possível averiguar se ataques de dia zero, já divulgados e que causaram prejuízos em algumas aplicações, causam danos no software testado. Caso o resultado seja positivo, será possível avisar o fornecedor da vulnerabilidade, para que atualizações sejam feitas.

5. Resultados esperados

A realização deste trabalho de estudo das vulnerabilidades descritas nas bases NVD, CVE e *A-Z Listing of Threats & Risk* e das combinações das mesmas pode contribuir para que ataques de dia zero, já divulgados, possam ser utilizados para prevenir vulnerabilidades em novas aplicações. Isso não significa que seja possível abrir mão de outros meios de prevenção, como de firewalls, sensores, teste de penetração, antivírus, atualizações de navegadores e softwares, para prevenção de ataques.

Referências

Bertucci, D. (2017). Segurança do data center em 2017.

CERT.br (2012). Cartilha de segurança para internet.

Evans, D. (2011). The internet of things: How the next evolution of the internet is changing everything.

Kaspersky, E. (2012). Os perigos dos exploits e dias zero e como preveni-los.

Last, D. (2016). Forecasting zero-day vulnerabilities. *11th Annual Cyber and Information Security Research Conference*, 13.

Leyla Bilge, T. D., editor (2012). *Before we knew it: an empirical study of zero-day attacks in the real world*. ACM.

McAffe (2012). Relatório da mcafee sobre ameaças: primeiro trimestre de 2012.

McCarthy, L. (2004). Ataques desconhecidos requerem sistemas avançados de aviso.

MITRE (2017). About CVE.

NIST (2017). National vulnerability database.

Paolo Foti, Jart Armin, M. C., editor (2015). *0-Day Vulnerabilities and Cybercrime*. IEEE.

Pedrosa, F. (2007). IntroduÇÃo aos exploits.

Pratap Kumar, R. K. S. (2016). A review on 0-day vulnerability testing in web application. *2nd International Conference on Information and Communication Technology for Competitive Strategies*, 141.

Santos, G. M. D. (2017). Bases de vulnerabilidades.

Serge Egelman, Cormac Herley, P. C. v. O., editor (2013). *Markets for Zero-Day Exploits: Ethics and Implications*. ACM.

Symantec. Listing of threats & risks.