

Concepção e Implementação de Modelos de Machine Learning usando Árvores de Decisão

Henrique Faria and Paulo Bento

Universidade do Minho, Mestrado Integrado em Engenharia Informática

Resumo Palavras-chave: Tratamento de dados · Tunning

1 Q5 - Resultados Globais

Após o scan feito pelo OpenVass obtivemos os seguintes dados.

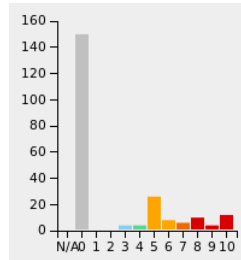


Figura 1. Gravidade das vulnerabilidades encontradas

Nesta figura podemos observar o número de vulnerabilidades conhecidas e atualmente na base de dados do OpenVass encontradas no Metasploite2. Em seguida apresentam-se as vulnerabilidades encontradas no Sistema visado.

1.1 Apache HTTP Server

O servidor avaliado corre com um servidor Apache HTTP e está vulnerável a permitir acesso a informação sensível através das cookies. O error ocorre devido á resposta de erro por defeito com código de estado 400, quando não é configurado um documento personalizado de erro.

Isto pode levar a que um atacante consiga obter informação sensível que possa auxiliar num ataque futuro.

Pode ser facilmente corrigido atualizando para uma versão 2.2.22 ou posterior.

1.2 phpMyAdmin

O servidor está a correr phpMyAdmin e é vulnerável a cross-site scripting. Isto permite que atacantes conduzam ataques com injeção de código HTML arbitrário para gerar ataques de phishing.

Ainda não foi criada uma solução e provavelmente nenhuma será criada. Deve-se mudar o serviço ou desabilitar a resposta.

1.3 Samba MS-RPC

Esta vulnerabilidade permite que atacantes executem comandos arbitrários na shell. Com isto o atacante pode correr comandos na shell com as permissões da aplicação.

Para corrigir isto basta fazer uma atualização do software usado.

1.4 PostgreSQL

Podia-se aceder a uma base de dados PostgreSQL ao usar credenciais fracas, nomeadamente com a password "postgres".

Para evitar acessos indevidos deve-se redefinir a palavra pass o mais cedo possível.

1.5 VNC Brute Force Login

Este método passa por tentar aceder como uma password dada via protocolo VNC, a password usada é password.

Basta substituir a password por uma mais difícil.

1.6 DistCC

Esta vulnerabilidade passa por aproveitar a falta de restrições nos acessos às portas do servidor, dado que o DistCC confia cegamente nos clientes. Um atacante pode simplesmente correr comandos arbitrários no servidor.

Este problema resolve-se fazendo uma atualização do software.

1.7 Distributed Ruby

Esta falha permite que sistemas não autorizados executem comandos distribuídos, isto pois o Distributed Ruby não previne atividades de acesso privilegiado, caso este corra com acesso privilegiado um atacante pode executar comandos ou scripts ruby.

Para colmatar a falha basta restringir as permissões do serviço caso se permita o acesso a utilizadores não confiáveis ou então definir ACLs apropriadas no sistema.

1.8 Ingreslock

Uma backdoor é instalado no servidor remoto. O serviço responde a um id: uid=0, gid=0. Com isto um atacante pode executar código arbitrário com privilégios "root".

Não há correção disponível para esta vulnerabilidade.

2 Q6 - Tráfego Anómalo

Nesta secção serão analisados dois exemplos de tráfego anómalo reportados pelo Snort.

2.1 PROTOCOL - SNMP AgentX/tcp request

O 5º pacote identificado pelo Snort corresponde ao pacote número 233 do Wireshark.

Este pacote corre sobre TCP e tem origem no endereço: 172.16.1.128, porta: 46754 destinando-se ao endereço: 172.16.1.129, porta: 80.

– CVSS Scores & Vulnerability Types	
CVSS Score	10.0
Confidentiality Impact	Complete (There is total information disclosure, resulting in all system files being revealed.)
Integrity Impact	Complete (There is a total compromise of system integrity. There is a complete loss of system protection, resulting in the entire system being compromised.)
Availability Impact	Complete (There is a total shutdown of the affected resource. The attacker can render the resource completely unavailable.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	Admin
Vulnerability Type(s)	Denial Of Service Gain privileges
CWE ID	264

Figura 2. CVE-2002-0012

CVE-2002-0012 Definição: Há vulnerabilidades num grande número de implementações do SNMP que permitem que atacantes possam efetuar ataques de denial of Service ou ganhar privilégios via uma armadilha SNMPv1.

Modo de proceder: O atacante provoca um erro cujo estado seja 400. Devido a um erro na criação de um documento personalizado este pode ser explorado para expor "httpOnly"cookies.

2.2 FTP Command Overflow Attempt

O 13º pacote identificado pelo Snort corresponde ao pacote número 9484 do WireShark.

Este pacote corre sobre TCP fazendo uso de FTP e tem origem no endereço: 172.16.1.128, porta: 56385 destinando-se ao endereço: 172.16.1.129, porta: 21.

De seguida apresenta-se o CVE mais recente relacionado com este problema.

– CVSS Scores & Vulnerability Types	
CVSS Score	6.5
Confidentiality Impact	Partial (There is considerable informational disclosure.)
Integrity Impact	Partial (Modification of some system files or information is possible, but the attacker does not have control over what can be modified, or the scope of what the affect is limited.)
Availability Impact	Partial (There is reduced performance or interruptions in resource availability.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)
Authentication	Single system (The vulnerability requires an attacker to be logged into the system (such as at a command line or via a desktop session or web interface).)
Gained Access	User
Vulnerability Type(s)	Execute Code Overflow
CWE ID	CWE id is not defined for this vulnerability

Figura 3. CVE-2002-0012

CVE-2007-0019 Definição: O atacante executa código arbitrário fazendo uso de um comando LIST longo e outros pedidos ao serviço FTP permitindo que estes executem código arbitrário com pedidos não especificados ao serviço HTTP.

3 Q7 - Vulnerabilidades Snort vs OpenVass

- A principal razão para o Snort apresentar vulnerabilidades que o openVas não reporta deve-se aos falsos positivos.
Há comportamentos reconhecidos como tráfego anómalo, por exemplo um "port scan" feito por um administrador da rede. O Snort vai detetar este tráfego e sinaliza-lo como tal, no entanto não se trata de uma tentativa de intrusão. Já o OpenVass não reporta este scan visto que apenas reporta falhas anómalas como por exemplo a utilização do serviço regexd sem ser sobre ssh que permite o envio de passwords em texto limpo.
- Uma razão pela qual o Snort pode apresentar mais falhas do que o OpenVas trata-se de o OpenVas trabalhar sobre aplicações e o Snort sobre a camada de rede. Assim o Snort pode reportar vários pacotes com comportamento anómalo como um "port scan", no entanto estes pacotes não representam necessariamente a tentativa de exploração de uma falha pois por exemplo este scan pode ter sido feito pelo administrador da rede. Já o OpenVass deteta e reporta uma falha numa aplicação ou serviço no qual este fornece de alguma forma manipulação ou obtenção de informação sensível.

4 Q8 - Correção de vulnerabilidades

Para visualização da correção das falhas alvo, note-se que o OpenVass faz um scan seguindo sempre a mesma ordem de teste portanto após a indicação de como foi corrigida cada falha aparecerá uma imagem com o antes e o depois das correções e vendo a falha anterior á falha alvo e a posterior constatar-se-á que a falha a ser corrigida foi de facto colmatada.

4.1 HTTP Debugging Methods (Trace/Track) Enabled

- Descrição da Vulnerabilidade: O servidor Web permite metodos de rastreamento HTTP que são usados para corrigir conexões ao servidor web. Neste caso o método ativo é o TRACE. Com esta vulnerabilidade o atacante pode enganar o sistema fazendo com que este lhe envie as suas credencias.
- Método de resolução: Basta desabilitar o uso do Trace para corrigir esta vulnerabilidade.

Para proceder á correção do problema primeiro foi preciso encontrar o ficheiro do apache2 responsável por desabilitar o trace, este ficheiro chama-se *httpd.conf* e está localizado em */etc/apache2/*.

Em seguida abriu-se este ficheiro recorrendo ao comando nano como super-user: *sudo nano httpd.conf* e escreveu-se o seguinte no ficheiro: *TraceEnable Off*.

HTTP Security Headers Detection		80%	172.16.1.129	80/tcp	Mon Nov 25 19:31:53 2019
HTTP Debugging Methods (TRACE/TRACK) Enabled		99%	172.16.1.129	80/tcp	Mon Nov 25 19:34:08 2019
phpinfo() output Reporting		80%	172.16.1.129	80/tcp	Mon Nov 25 19:33:52 2019

Figura 4. HTTP Debugging Methods (Trace/Track) Enabled

HTTP Security Headers Detection		80%	172.16.1.129	80/tcp	Tue Nov 26 04:02:48 2019
phpinfo() output Reporting		80%	172.16.1.129	80/tcp	Tue Nov 26 04:05:29 2019

Figura 5. HTTP Debugging Methods (Trace/Track) Disabled

- Vulnerabilidades extra corrigidas: Nenhuma.

4.2 Rexecd Service Detection

- Descrição da Vulnerabilidade: Este serviço permite a execução de comandos na shell de um computador remoto. No entanto o rexec permite a autenticação lendo o username e password descriptados da socket.
- Método de resolução: Desabilitar o uso do serviço rexec e usar alternativas como o SSH.

Para realizar isto temos de ir até à pasta /etc, nesta abrimos com o comando *sudo nano inetd.conf* o ficheiro inetd.conf e alteramos a linha: *exec stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.rexecd* para *exec stream tcp nowait root /usr/sbin/sshd /usr/sbin/in.rexecd*.

Isto fará com que este serviço inicie o servidor rexecd usando ssh.

Telnet Service Detection		80%	172.16.1.129	23/tcp	Mon Nov 25 19:21:43 2019
rexec Passwordless / Unencrypted Cleartext Login		80%	172.16.1.129	512/tcp	Mon Nov 25 19:32:32 2019
PostgreSQL Detection		80%	172.16.1.129	5432/tcp	Mon Nov 25 19:19:04 2019

Figura 6. Rexecd Service over tcpd



Telnet Service Detection		80%	172.16.1.129	23/tcp	Tue Nov 26 03:54:25 2019
PostgreSQL Detection		80%	172.16.1.129	5432/tcp	Tue Nov 26 03:52:16 2019

Figura 7. Rexecd Service over sshd

- Vulnerabilidades extra corrigidas: Nenhuma.

4.3 Bind Shell Backdoor Detection

- Descrição da Vulnerabilidade: Há uma possível backdoor instalada no servidor remoto. O comando está a responder a um id=0(root) e gid=0(root). Um atacante pode executar um comando no contexto da aplicação e comprometer o sistema.
- Método de resolução: Existem 3 métodos de corrigir esta vulnerabilidade.
- Verificar se o servidor foi comprometido e reinstalar o sistema se necessário.
- Desativar o ingreslock.

O método aplicado é o bloqueio do início do serviço simplesmente comentando a linha *ingeslock stream tcp nowait root /bin/bash bash -i*.

PHP-CGI-based setups vulnerability when parsing query string parameters from php files.		95%	172.16.1.129	80/tcp	Mon Nov 25 19:36:51 2019
Possible Backdoor: Ingreslock		99%	172.16.1.129	1524/tcp	Mon Nov 25 19:38:19 2019
PostgreSQL weak password		99%	172.16.1.129	5432/tcp	Mon Nov 25 19:39:53 2019

Figura 8. Ingreslock backdoor enabled



PHP-CGI-based setups vulnerability when parsing query string parameters from php files.		95%	172.16.1.129	80/tcp	Tue Nov 26 04:08:40 2019
PostgreSQL weak password		99%	172.16.1.129	5432/tcp	Tue Nov 26 04:11:45 2019

Figura 9. Ingreslock Service not started

- Vulnerabilidades extra corrigidas: Nenhuma.

4.4 VNC Server ‘password’ Password

- Descrição da Vulnerabilidade: O servidor VNC a correr no computador remoto utiliza uma password fraca, por defeito esta password é ‘password’. Um atacante pode tirar partido disto para obter controlo sobre o sistema.

Note-se que antes de podermos corrigir o a falha do VNC tivemos de o inicializar com o comando: `sudo vncserver`. Isto pois o VNC estava mal inicializado, posteriormente foi criado um ficheiro `passwd` em `/home/msfadmin/.vnc/` onde foi colocada a nova password do vnc.

- Método de resolução: Alterar a palavra pass para uma forte.

Para proceder á geração de uma password forte utilizamos um gerador online com 16 caracteres. A password obtida foi: `5?V=X#&kAdB'H6+y`, infelizmente o vnc tem as password truncadas para 8 caracteres automaticamente, assim a password usada foi: `5?V=X#&k`.

Nota: Como as passwords do VNC são truncadas para 8 caracteres, este é extremamente inseguro. Como o objetivo do trabalho é corrigir o mesmo apenas mudamos a password, no entanto achamos melhor que seja usado outro serviço e que este seja descontinuado.

OS Detection Consolidation and Reporting		80%	172.16.1.129	general/tcp	Mon Nov 25 19:32:38 2019
VNC Brute Force Login		95%	172.16.1.129	5900/tcp	Mon Nov 25 19:38:34 2019

Figura 10. VNC Server com a password ‘password’

Vulnerability		Severity	QoD	Host	Location	Created
SSL/TLS: Certificate Signed Using A Weak Signature Algorithm		4.0 (Medium)	80%	172.16.1.129	5432/tcp	Tue Nov 26 08:51:52 2019
OS Detection Consolidation and Reporting		0.0 (Log)	80%	172.16.1.129	general/tcp	Tue Nov 26 08:57:12 2019
VNC Brute Force Login		0.0 (Log)	95%	172.16.1.129	5902/tcp	Tue Nov 26 09:01:32 2019

Figura 11. VNC Server com a password ‘5?V=X#&k’






Vulnerability		Severity		QoD	Host	Location	Actions
VNC Brute Force Login		0.0 (Log)		95%	172.16.1.129	5902/tcp	 
Summary Try to log in with given passwords via VNC protocol.							
Vulnerability Detection Result Too many unsuccessful connection attempts are made which means the scanner IP got blocked. Therefore the brute force check was aborted.							

Figura 12. Resultado do acesso ao VNC Server com a password '5?V=X#&k'

- Vulnerabilidades extra corrigidas: Nenhuma
- Vulnerabilidades criadas: A correção da vulnerabilidade propósta levou ao aparecimento de outra vulnerabilidade: *DNS*, como se vê na figura acima.