REPORT DE VULNERABILIDADE

MATHEUS MONTEIRO DE ALMEIDA

RIO DE JANEIRO

2024

Esse report é baseado em uma análise manual, utilizando ferramentas para análise de IPs e hosts públicos (VirusTotal) e scripts em Python, com a biblioteca Pandas. Esses scripts foram gerados pelo ChatGPT e/ou por mim, com ajustes de acordo com a minha necessidade. Fora isso, todas as outras afirmações e sugestões foram feitas por mim.

Análise geral do arquivo raw .CSV:

Normalmente, quando preciso fazer uma análise de tráfego de rede, o que recebo geralmente é um arquivo de PCAP, onde a análise acaba sendo automatizada pelo IDS open source que uso (Suricata), com as regras específicas que criei ou obtive na internet de organizações que disponibilizam essas regras gratuitamente (EmergingThreats). Porém, esse não é o caso, então vamos realizar uma análise manual.

Utilizando o script **topvalues.py** (gerado por mim/ChatGPT), consegui buscar os valores mais frequentes de cada coluna deste CSV e fui direto verificar os top hosts e seus respectivos paths.

Top valores para ClientRequestHost:

ClientRequestHost	Ocorrências
porter.biz	13.232
acosta.com	4.815
joseph.org	2.262
brown.net	2.088
pearson.com	1.621

O próximo passo foi analisar as portas utilizadas.

ClientSrcPort:

ClientSrcPort	Ocorrências
0	34
41568	8
42460	7
55798	7
34332	7

Depois de verificar que os hosts em si não tinham flags no **VirusTotal** de serem maliciosos e que as portas são de numeração alta, sem uso específico catalogado no **speedguide.net,** algo que chamou a atenção foi os países para onde as requisições estavam sendo feitas.

Top valores para ClientCountry:

ClientCountry	Ocorrências
in	18.372
us	11.481
br	37
jp	34
gb	26

O fato de haver tanta conexão com dois países tão distintos é suspeito. A não ser que seja uma empresa dos EUA com call center na Índia (ou vice-versa), isso pode indicar uma provável tentativa de acesso remoto, talvez já efetuada com sucesso, possivelmente por algum malware que obteve uma backdoor na máquina ou até mesmo uma infecção no sistema para mineração.

Outro ponto importante é a quantidade de comunicação feita via HTTP. Nesse caso eu utilizei o **tophttp.py**, os hosts e os IPs não tiveram flags de maliciosos no VirusTotal.

Porém, pelo fato de terem sido feitas requisições GET e principalmente POST (mesmo que poucas), é provável que algum tipo de dado tenha sido inserido pelo usuário (cliente) nesses hosts (servidores), seja para compra utilizando dados de cartão ou criação de usuário. Fazer isso via HTTP é um risco absurdo, pois qualquer pessoa com o mínimo de conhecimento de frontend ou de pentest web pode acessar ou visualizar informações críticas desses hosts. Pior ainda, qualquer pessoa que esteja usando um Wireshark na rede pode ver esses dados críticos sem criptografia nenhuma.

Top valores para ClientIP (HTTP):

ClientIP	Ocorrências
113.50.114.50	8
19.208.152.96	3
196.136.28.168	3
25.195.20.94	3
135.130.83.6	3

Top valores para ClientRequestHost (HTTP):

ClientRequestHost	Ocorrências
walters-thompson.com	7
pruitt.com	5
snyder.com	5
yang-herrera.biz	4
rodriguez.com	4

Agora, provavelmente o mais alarmante foi o que encontrei no **Top valores para ClientRequestPath** em tráfego **HTTP**:

Top valores para ClientRequestPath (HTTP):

ClientRequestPath	Ocorrências
/small/day/nation	42
/certainly/next	6
/discuss/safe/network	3
/search?q=<script>alert(0)</script>	**3 ←—--**
/protect	2

Claramente, há uma tentativa de explorar uma vulnerabilidade de **XSS**. Assim que vi, pensei que fosse via **SSRF**, mas, como a requisição não tem como alvo claro um diretório do sistema, é XSS mesmo.

Após identificar três pontos críticos nessa análise de tráfego, já é mais que suficiente recomendar a utilização de ferramentas de proteção e análise de tráfego.

Pontos críticos identificados:

1. **Tráfego em grande quantidade para países adversos e não correlacionados (Índia e EUA)**.
2. **Tráfego razoável em protocolo inseguro (HTTP)**.
3. **Requisição clara para tentativa de explorar vulnerabilidade de XSS**.

Existe a possibilidade de encontrar mais informações importantes nesse arquivo CSV relacionadas a vulnerabilidades. Entretanto, já é mais que suficiente partirmos para a parte de implantação de soluções.

Soluções recomendadas:

1. **Servidor DNS próprio (ex.: Pi-hole)**:
    - Com um servidor DNS próprio, é possível configurar bloqueios para URLs conhecidas e indesejáveis, evitando que os usuários da rede acessem URLs já sinalizadas como maliciosas.
2. **IDS para análise de tráfego (ex.: Suricata)**:
    - O Suricata é uma ferramenta excelente para análise de tráfego. Como IDS, é possível gerar diversos tipos de alerta, desde acessos a ranges de IPs públicos indesejados até execução de scripts desconhecidos na rede.
3. **Firewall ou IPS com regras específicas (ex.: Fortinet/Palo Alto/PfSense)**:
    - Proteção total do tráfego com regras predeterminadas, evitando qualquer tipo de comunicação indesejada.

Automação:

Nos repositórios disponíveis, existem arquivos **docker-compose.yml** para facilitar a aplicação das soluções recomendadas, respectivamente para o Pi-hole e o Suricata, utilizando Docker para uma implementação simplificada.

(Obviamente o arquivo precisa ser alterado de acordo com as necessidades de cada um e customizado posteriormente)