BOLETIM DE SEGURANÇA

Malware Loader Latrodectus ganha destaque em campanhas de phishing, substituindo o IcedID

heimdall
security research
A DIVISION OF ISH

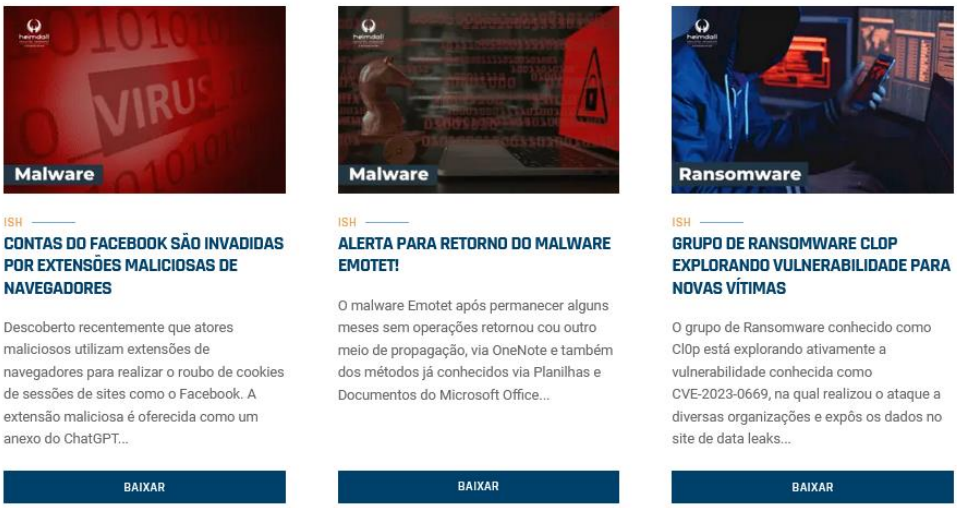
TLP: CLEAR

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

Recentemente foi observado por pesquisadores de segurança cibernética da Elastic, um aumento nas campanhas maliciosas de phishing entregando o malware **Latrodectus**, um malware do tipo Loader que se acredita ser o sucessor do malware IcedID. Geralmente, essas campanhas seguem um padrão de infecção bem conhecido, onde arquivos JavaScript extensos são utilizados para acionar o msiexec.exe através do WMI. Isso resulta na instalação de um arquivo MSI hospedado remotamente, frequentemente encontrado em um compartilhamento WEBDAV.

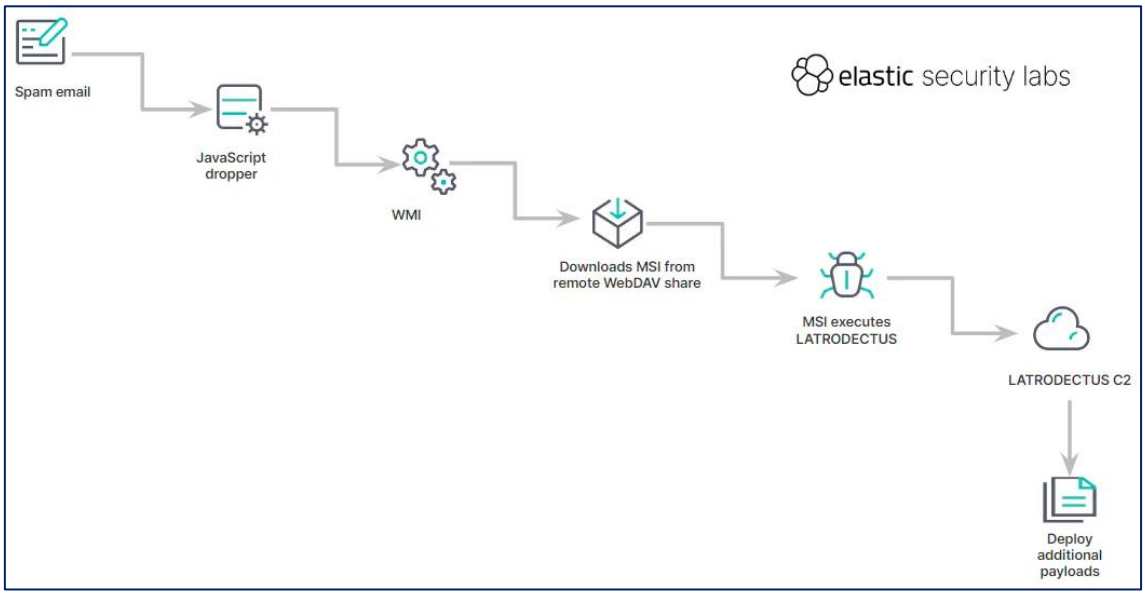


Figura 1 – Cadeia de infecção para entrega do Latrodectus.

2 ANÁLISE DE AMOSTRA DO MALWARE LATRODECTUS

Neste exemplo apresentado pelos pesquisadores o malware Latrodectus veio inicialmente disfarçado com informações de arquivo como um componente do driver de modo kernel do Bitdefender (TRUFOS.SYS).

File Version Information

Copyright	Copyright © Bitdefender
Product	Bitdefender Antivirus
Description	Trufos API
Original Name	TRUFOS.DLL
Internal Name	TRUFOS.DLL
File Version	2.5.4.62.761d05c Free Build

Figura 2 – Informações iniciais sobre a amostra identificada.

O malware é uma DLL com 4 exportações diferentes, e cada exportação recebe o mesmo endereço de exportação, conforme mostra imagem abaixo.

Exports

Name	Address	Ordinal
extra	0000000180003CE4	1
follower	0000000180003CE4	2
run	0000000180003CE4	3
scub	0000000180003CE4	4
DllEntryPoint	**0000000180003C7C**	**[main entry]**

Figura 3 – Exportações para o malware.

Todas as strings dentro do malware são protegidas usando um algoritmo simples nos bytes criptografados e aplicando uma transformação executando operações aritméticas e bit a bit.

```
def decrypt_string(encrypted_bytes: bytes) -> bytes:
    x = cast.u32(encrypted_bytes[:4])
    y = cast.u16(encrypted_bytes[4:6])
    byte_size = cast.u16(cast.p32(x ^ y)[:2])
    decoded_bytes = bytearray(byte_size)

    for i, b in enumerate(encrypted_bytes[6 : 6 + byte_size]):
        decoded_bytes[i] = ((x + i + 1) ^ b) % 256

    return bytes(decoded_bytes)
```

Figura 4 – Strings do malware.

Latrodectus ofusca a maioria de suas importações até o tempo de execução. No início do programa, ele utiliza o PEB em conjunto com uma soma de verificação CRC32 para resolver os módulos kernel32.dll e ntdll.dll, bem como

suas funções. Para resolver bibliotecas adicionais, como user32.dll ou wininet.dll, o malware adota uma abordagem diferente, realizando uma pesquisa curinga (*.dll) no diretório do sistema Windows. Ele extrai cada nome de arquivo DLL e os submete diretamente a uma função de soma de verificação CRC32.

```
des::GetSystemDirectory();
lpFileName = des::GetSystemDirectory();
if ( !lpFileName )
  return 0i64;
des::DecryptString(dword_180010088, str_wildcard_dll);// \*.dll
_str_wildcard_dll = str_wildcard_dll;
if ( !des::CombinePath(&lpFileName, str_wildcard_dll) )
  return 0i64;
LibraryW = 0i64;
des::ZeroOutMemory(&FindFileData, 0x250ui64);
hFindFile = FindFirstFileW(lpFileName, &FindFileData);
if ( hFindFile != -1i64 )
{
  while ( FindNextFileW(hFindFile, &FindFileData) && hFindFile != -1i64 )
  {
    v2 = des::CountLengthWideStr(FindFileData.cFileName);
    crc32_hash = des::checksum::CRC32(FindFileData.cFileName, 2 * v2);
    if ( crc32_hash == hash )
    {
      LibraryW = LoadLibraryW(FindFileData.cFileName);
      break;
    }
  }
}
des::FreeUpMemoryViaSyscall(lpFileName);
return LibraryW;
```

Figura 5 – Pesquisa de DLL usando uma soma de verificação CRC32.

Após resolver todas as importações, o malware executa várias verificações anti-análise. A primeira delas monitora um depurador, procurando pelo sinalizador BeingDebugged dentro do Process Environment Block (PEB). Se um depurador for detectado, o programa será terminado.

```
__int64 des::BeingDebuggedCheck()
{
  return GetPEB()->BeingDebugged;
```

Figura 6 – Checagem BeingDebugged via PEB.

Para contornar sandboxes ou máquinas virtuais, que geralmente têm um número limitado de processos ativos, são empregadas duas verificações de validação. Essas verificações comparam o número de processos em execução com a versão do sistema operacional para determinar se há discrepâncias.

```
os_version = des::RetrieveMajorMinorOSVersions();
if ( des::GetNumberofCurrentProcessesRunning() < 75 && os_version >= 6 )
  return 0xFFFFFFFFi64;
if ( des::GetNumberofCurrentProcessesRunning() < 50 && os_version < 6 )
  return 0xFFFFFFFFi64;
```

Figura 7 – Processos e verificações de Sistema Operacional.

```
if ( RtlGetVersion )
  RtlGetVersion(&os_ver);
if ( !RtlGetVersion )
  GetVersionExW(&os_ver);
if ( os_ver.dwMajorVersion != 5 || os_ver.dwMinorVersion )
{
  if ( os_ver.dwMajorVersion == 5 && os_ver.dwMinorVersion )
  {
    return 1;
  }
  else if ( os_ver.dwMajorVersion != 6 || os_ver.dwMinorVersion )
  {
    if ( os_ver.dwMajorVersion == 6 && os_ver.dwMinorVersion == 1 )
    {
      return 3;
    }
    else if ( os_ver.dwMajorVersion == 6 && os_ver.dwMinorVersion == 2 )
    {
      return 4;
    }
    else if ( os_ver.dwMajorVersion == 6 && os_ver.dwMinorVersion == 3 )
    {
      return 5;
    }
    else if ( os_ver.dwMajorVersion != 10 || os_ver.dwMinorVersion )
    {
      if ( os_ver.dwMajorVersion == 10 && !os_ver.dwMinorVersion && os_ver.dwBuildNumber >= 0x55F0 )
        return 7;
    }
    else
    {
      return 6;
    }
```

Figura 8 – Enumeração de Sistema Operacional.

Essas duas condições se resumem da seguinte maneira:

- Latrodectus será terminado se o número de processos for inferior a 75 em conjunto com uma versão recente do sistema operacional, como Windows 10, Windows Server 2016 ou Windows 11.
- Latrodectus será encerrado se o número de processos for inferior a 50 em conjunto com uma versão mais antiga do sistema operacional, tal como Windows Server 2003 R2, Windows XP, Windows 2000, Windows 7, Windows 8 ou Windows Server 2012/R2.

Depois de verificar o ambiente de sandbox, o LATRODECTUS também analisa se o processo atual está sendo executado no WOW64, que é um subsistema dos sistemas operacionais Windows. O WOW64 permite que aplicativos de 32 bits sejam executados em sistemas de 64 bits. Se essa condição for verdadeira (ou seja, se estiver sendo executado como um aplicativo de 32 bits em um sistema operacional de 64 bits), o malware será encerrado.

```
CurrentProcess = GetCurrentProcess();
IsWow64Process(CurrentProcess, &Wow64Process);
if ( Wow64Process )
  return 0xFFFFFFFFi64;
```

Figura 9 – Verificação do processo IsWow64.

A última etapa de verificação envolve a análise do endereço MAC utilizando a chamada GetAdaptersInfo() da iphlpapi.dll. Se não for detectado um endereço MAC válido, o malware será encerrado.

```
AdapterInfo = 0i64;
v4 = 0i64;
SizePointer = 0;
AdaptersInfo = GetAdaptersInfo(0i64, &SizePointer);
if ( AdaptersInfo == ERROR_BUFFER_OVERFLOW )
{
  AdapterInfo = des::AllocateMemoryViaSyscall(SizePointer);
  AdaptersInfo = GetAdaptersInfo(AdapterInfo, &SizePointer);
  while ( AdapterInfo->AddressLength <= 6 )
  {
    AdapterInfo = AdapterInfo->Next;
    if ( !AdapterInfo )
      goto LABEL_6;
  }
  return 0i64;
```

Figura 10 – Verificação de endereço MAC.

Configuração de persistência do malware

O malware criará um caminho de pasta conforme especificado por um parâmetro de configuração. Este parâmetro determina onde o Latrodectus será armazenado no disco, podendo incluir os seguintes diretórios:

- *AppData*
- *Desktop*
- *Startup*
- *Personal*
- *Local\AppData*

Autoexclusão do malware

Uma característica notável do malware é sua capacidade de autoexclusão, uma técnica descoberta pelos pesquisadores Jonas Lykkegaard e implementada por Lloyd Davies no repositório delete-self-poc. Essa técnica permite que o Latrodectus se exclua enquanto ainda está em execução, utilizando um fluxo de dados alternativo. O Elastic Labs observou essa técnica sendo adotada em malwares, como na família de ransomware ROOK. O objetivo provável é dificultar os processos de resposta a incidentes, interferindo na coleta e análise de dados. O malware compilado contém uma string (:wtfbbq) encontrada no repositório mencionado.

```
if ( SetFileInformationByHandle(hFile, FileRenameInfo, fRename, str_wtfbbq_plus_space[0]) )
{
  des::FreeUpMemoryViaSyscall(frename);
  CloseHandle(hFile);
  hFile = CreateFileW(lpFileName, DELETE, 0, 0i64, OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL, 0i64);
  if ( hFile == -1i64 )
  {
    return 0xFFFFFFFFi64;
  }
  else
  {
    FileInformation.DeleteFileA = 0;
    des::MemSetToZero(&FileInformation, 1ui64);
    FileInformation.DeleteFileA = 1;
    if ( SetFileInformationByHandle(hFile, FileDispositionInfo, &FileInformation, 1u) )
    {
      CloseHandle(hFile);
      return 0i64;
    }
```

Figura 11 – Parte do código para a autoexclusão do malware.

O Latrodectus é um malware avançado que surgiu recentemente e é conhecido por suas capacidades sofisticadas de roubo de informações e controle remoto. Este malware é capaz de capturar credenciais, registrar pressionamentos de teclas, tirar capturas de tela e exfiltrar dados confidenciais. Ele se propaga principalmente através de campanhas de phishing e utiliza técnicas de evasão para evitar detecção por softwares antivírus. Latrodectus também pode instalar backdoors para acesso contínuo ao sistema comprometido e é frequentemente utilizado em ataques direcionados contra organizações de alto valor, como instituições financeiras e empresas de tecnologia. Sua versatilidade e capacidade de adaptação o tornam uma ameaça significativa no cenário de cibersegurança.

3 MITRE ATT&CK - TTPs

Tática	Técnica	Detalhes
Execution	T1059.007 T1047 T1059.003	Latrodectus é entregue por meio de arquivos Javascript grandes, em média com mais de 800 KB preenchidos com texto aleatório. O Dropper Javascript invoca WMI para montar um compartilhamento WEBDAV e invoca msiexec para instalar um arquivo msi remoto. ID de comando Latrodectus - Coleta informações do sistema por meio de uma série de execução de cmd.exe.
Defense Evasion	T1027 T1218.007 T1218.011 T1070.004	O malware é entregue com arquivos ou informações ofuscadas Arquivo MSI hospedado em Webdav remoto e executado em modo silencioso. Uma vez executado, ele descarta uma DLL e inicia rundll32 para carregá-la por meio do binário Advanced installer viewer.exe. Rundll32 carrega a DLL Latrodectus do AppData e inicia a injeção de código. Parte do comando de autoatualização de DLL de malware e também quando a DLL não está sendo executada a partir do AppData, o Latrodectus se excluirá durante a execução e reiniciará a partir do novo caminho ou executará uma versão atualizada de si mesmo aproveitando esta técnica.
Defense Evasion Privilege Escalation	T1055	A execução do Shellcode aciona três alertas de comportamento de endpoint e um alerta de detecção de ameaça à memória.
Execution Persistence Privilege Escalation	T1053.005	O malware pode persistir usando tarefas agendadas (rundll32 criará uma tarefa agendada)

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:

Atualizações de software e patches

- Mantenha todos os sistemas operacionais, softwares e aplicativos atualizados. O Malware Latrodectus frequentemente explora vulnerabilidades conhecidas, que são corrigidas pelos fabricantes através de patches e atualizações.

Soluções antivírus e anti-malware

- Utilize soluções robustas de antivírus e anti-malware com proteção em tempo real. Configure essas soluções para realizar varreduras regulares e garantir que elas sejam atualizadas frequentemente.

Treinamento em conscientização de segurança

- Eduque funcionários e usuários sobre as melhores práticas de segurança, incluindo a identificação de phishing e outras táticas comuns usadas para disseminar malware. Realize treinamentos regulares e simulações de ataques de phishing.

Controle de acesso e privilegiado

- Implemente o princípio de menor privilégio (PoLP), garantindo que os usuários tenham apenas os acessos necessários para realizar suas funções. Utilize autenticação multifatorial (MFA) para adicionar uma camada extra de segurança.

Backup e recuperação

- Mantenha backups regulares e testados de dados importantes, armazenando-os em uma localização segura, preferencialmente desconectada da rede principal. Isso minimiza o impacto de um ataque bem-sucedido, permitindo a rápida restauração dos dados.

Segurança de rede

- Utilize firewalls, gateways de e-mail seguros e outras tecnologias para monitorar e controlar o tráfego de rede. Implemente segmentação de rede para limitar a propagação de infecções dentro da organização.

Monitoramento e detecção de anomalias

- Implemente soluções de detecção e resposta de endpoint (EDR) e sistemas de detecção de intrusão para monitorar comportamentos anormais e possíveis sinais de comprometimento.

Resposta a incidentes

- Desenvolva e teste um plano de resposta a incidentes que inclua procedimentos para isolar dispositivos infectados, erradicar o malware, realizar forense para entender a extensão do comprometimento, e comunicar-se com as partes interessadas.

Auditoria e revisão

- Realize auditorias de segurança regulares e revisões de configurações para identificar e corrigir possíveis vulnerabilidades de segurança.

5 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:	da8ae8e1de522b20a462239c6893613e
sha1:	7f65ef885815d81d220f9f42877ff0d696b0134c
sha256:	aee22a35cbdac3f16c3ed742c0b1bfe9739a13469cf43b36fb2c63565111028c
File name:	TRUFOS.DLL

Indicadores de compromisso do artefato	
md5:	b4a482a7e96cfdef632a7af286120156
sha1:	73e3639a9388af84b9c0f172b3aeaf3823014596
sha256:	ead5ebf464c313176174ff0fdc3360a3477f6361d0947221d31287eeb04691b3
File name:	91b48f1d.msi

Indicadores de compromisso do artefato	
md5:	6682dc1281579bd8789a8d2c09ca4251
sha1:	67bb21c9665fc12d8dc6ef2ac775c3f6274bd0ed
sha256:	937d07239cbfee2d34b7f1fae762ac72b52fb2b710e87e02fa758f452aa62913
File name:	937d07239cbfee2d34b7f1fae762ac72b52fb2b710e87e02fa758f452aa62913.js

Indicadores de compromisso do artefato	
md5:	3be9e476da2e99adbc49591cbc94b4d9
sha1:	2155590f685d4e28c278123a1cca633e8746db78
sha256:	240677752d6ba09cc9f98275d694c500ed75808080fd6f8d750c16a526dc4ba7
File name:	240677752d6ba09cc9f98275d694c500ed75808080fd6f8d750c16a526dc4ba7

Indicadores de compromisso do artefato	
md5:	441ce23f19aa042727acff5787e06d9d
sha1:	d907481fbcb17743d312e2e9f1aa855a1abdd24c
sha256:	805b59e48af90504024f70124d850870a69b822b8e34d1ee551353c42a338bf7
File name:	.data

Indicadores de compromisso do artefato	
md5:	2e9a5b6ebc31fc7a6d750bec94a40ce6
sha1:	321e6e5276e9f75d763801e286420a6465015548
sha256:	d1e2e287c96c290e161c553d99a115e7d72f83f23c850621169a27cca936f51b
File name:	d1e2e287c96c290e161c553d99a115e7d72f83f23c850621169a27cca936f51b.exe

Indicadores de compromisso do artefato	
md5:	22f06d9fc9d95f0945fc6113c091a072
sha1:	672a5de375d84ac54ff1fc14ec65402c0abeca97
sha256:	465f931e8a44b7f8dff8435255240b88f88f11e23bc73741b21c20be8673b6b7

File name:	Update_72aa42a8.dll

Indicadores de compromisso do artefato	
md5:	d32db5208d83134ba5c8d6b8c8289aeb
sha1:	9866814c66431df3a6d96ac8c89ba535340f0ea0
sha256:	34aff1767909ff582d15949922549fddb5849f163260ad3efdc32d4f869fdf09
File name:	34aff1767909ff582d15949922549fddb5849f163260ad3efdc32d4f869fdf09.exe

Indicadores de compromisso do artefato	
md5:	54feebf7544cd0c82d019eed11dd3b2e
sha1:	c849ca34a04672104feeb176dcb148ba530ea9de
sha256:	38450cf934121c9f92785beffb736029190147523109607683240 29d9ba91e13
File name:	38450cf934121c9f92785beffb736029190147523109607683240 29d9ba91e13.exe

Indicadores de compromisso do artefato	
md5:	ff354796f9e0a1edea31b8c9f65cda1b
sha1:	eae04a4c6b9ea586e9d7f2a351de06e90d64a5bb
sha256:	7040402574a686f031c3af5fed37509d8979855397787aab70b2d1059099d2da
File name:	7040402574a686f031c3af5fed37509d8979855397787aab70b2d1059099d2da.exe

Indicadores de compromisso do artefato	
md5:	277c879bba623c8829090015437e002b
sha1:	897c609bbee39144798b400525bf8f59a51c2ff1
sha256:	a1e74120c32162d18c0245a8390360e9b63a11887e396c270e0ed35296952598
File name:	Update_3e0e709.dll

Tabela 2 – Indicadores de Compromissos de artefatos

Indicadores de URL, IPs e Domínios

Indicadores de URL, IPs e Domínios	
Domínio	aytobusesre[.]com scifimond[.]com gyxplonto[.]com neaachar[.]com workspacin[.]cloud gyxplonto[.]com malrgtrong[.]org bestfiveweb[.]com neaachar[.]com scifimond[.]com illoskanawer[.]com aytobusesre[.]com bewildering[.]org adaletli[.]org 1206jeans[.]com mvcpjotop[.]org hrlsgvir[.]org hyundaitmvbbla1[.]org tkcovmk[.]org necrtlr4[.]org unpeopled[.]org

	martialartshistory[.]org
IP	45[.]95[.]11[.]217 45[.]140.146[.]156 193[.]168[.]143.182 77[.]91[.]73[.]187 74.119.193.200 172[.]64[.]80[.]1 91[.]194[.]11[.]183 95[.]164[.]68[.]73 185[.]123[.]53[.]208 91[.]149[.]253[.]77 162[.]55[.]217[.]30 66[.]63[.]189[.]8 94[.]232[.]45[.]58

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.

6 REFERÊNCIAS

- **Heimdall *by* ISH Tecnologia**
- Elastic
- MITRE ATT&CK
- any.run

7 AUTORES

- **Ismael Pereira Rocha**

heimdall
security research

A DIVISION OF ISH