BOLETIM DE SEGURANÇA

Grupo Lazarus implantando novo Rat Kaolin em seus ataques

heimdall
security research
A DIVISION OF ISH

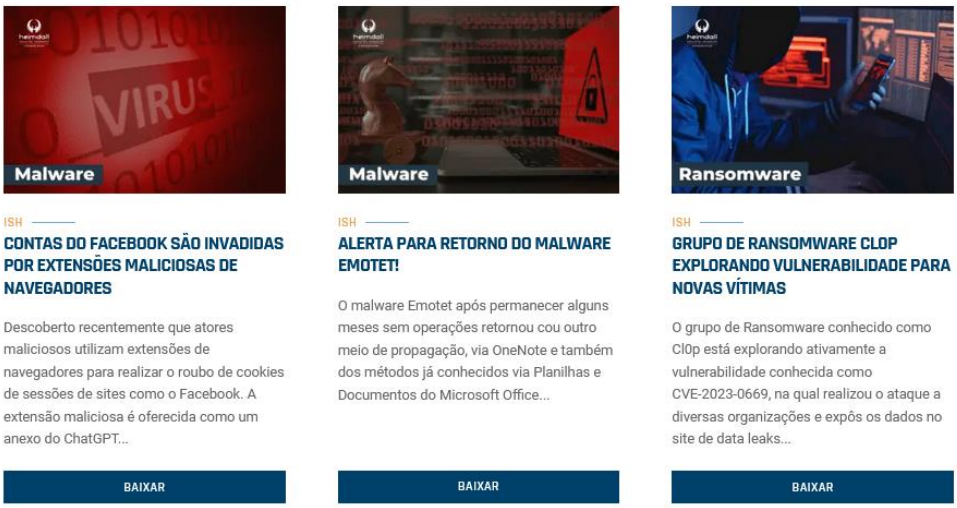
TLP: CLEAR

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

O grupo de cibercriminosos norte-coreano Lazarus Group realizou ataques seletivos na Ásia, utilizando o Kaolin RAT, um trojan de acesso remoto recém-desenvolvido. A estratégia incluiu o uso de iscas de emprego falsas, uma técnica comum do grupo. O ataque explorou a vulnerabilidade CVE-2024-21338 no driver appid.sys do Windows, visando neutralizar soluções de segurança.

2 DETALHES DO ATAQUE

Na fase preparatória dos ataques, o grupo adotou uma abordagem meticulosa na implantação de suas ferramentas, recorrendo ao uso de malware que não deixa arquivos residuais e criptografando suas ferramentas no disco rígido para ocultação.

O ataque se inicia com uma falsa proposta de trabalho, empregando engenharia social para ganhar a confiança do alvo. A plataforma de comunicação utilizada permanece incerta, indicando que LinkedIn, WhatsApp e e-mail são possíveis canais. O atacante então tenta enviar um arquivo ISO nocivo, camuflado como uma ferramenta VNC, parte do suposto processo seletivo. A partir do Windows 10, arquivos ISO podem ser montados automaticamente, facilitando o acesso ao seu conteúdo e potencialmente evitando o Mark-of-the-Web (**MotW**). Após estabelecer um vínculo com a vítima, o atacante a induz a montar o arquivo ISO, que contém **AmazonVNC.exe**, version.dll e **aws.cfg**, levando a vítima a executar o AmazonVNC.exe. Este arquivo, que se passa por um cliente VNC da Amazon, é na verdade o choice.exe, um aplicativo legítimo do Windows, usado aqui para carregar a DLL maliciosa version.dll, uma técnica conhecida como sideload.

A execução do AmazonVNC.exe resulta no carregamento da version.dll, que utiliza syscalls diretos para invocar funções nativas da API do Windows, buscando evitar detecção por técnicas defensivas. A DLL maliciosa, após adquirir os números de syscall corretos para a versão do Windows em uso, prepara-se para gerar um processo iexpress.exe, que hospedará uma carga maliciosa adicional do arquivo aws.cfg. Esta etapa de injeção ocorre somente se o antivírus Kaspersky estiver instalado, visando escapar de sua detecção. Caso contrário, o malware executa a carga diretamente. O arquivo aws.cfg, protegido pelo VMProtect, é capaz de baixar shellcode de um servidor de Comando e Controle (C&C), que suspeita-se ser um site legítimo invadido.

As investigações revelaram desafios ao tentar extrair o código shell do servidor C&C, devido à falta de resposta do URL malicioso. A análise telemétrica indicou ameaças em um cliente, correlacionando o carregamento de shellcode via arquivo ISO com a detecção do RollFling, um novo carregador não documentado. Além disso, o método de entrega do arquivo ISO e a presença do RollSling em máquinas comprometidas mostram semelhanças com táticas do grupo Lazarus. A instância do RollSling encontrada foi entregue pelo carregador RollFling, reforçando a conexão entre o shellcode não recuperado e o carregador inicial.

```
fix_api();
pFileName = (char *)&lpFileName;
if ( lpFileName._Myres >= 0x10 )
  pFileName = lpFileName._Bx._Ptr;
FirstFile = FindFirstFileExA(pFileName, FindExInfoStandard, &FindFileData, FindExSearchNameMatch, 0LL, 0);
if ( FirstFile != (HANDLE)-1LL )
{
  while ( FindFileData.cFileName[0] == '.'
       && (!FindFileData.cFileName[1] || FindFileData.cFileName[1] == '.' && !FindFileData.cFileName[2])
       || (FindFileData.dwFileAttributes & 0x10) != 0
       || load_binary_to_memory_and_execute_StartAction_export_function(FindFileData.cFileName) )
  {
    if ( !FindNextFileA(FirstFile, &FindFileData) )
      goto looking_in_another_path;
  }
  goto exit;
}
looking_in_another_path:
if ( !load_binary_to_memory_and_execute_StartAction_export_function(0LL) )
  v1 = 0;
v3 = v1;
if ( FirstFile != (HANDLE)-1LL )
exit:
  FindClose(FirstFile);
```

Figura 1 – Código RollSling.

O carregador RollFling, uma DLL mal-intencionada, é configurado como um serviço para garantir persistência no sistema da vítima. Ele é acompanhado por arquivos vitais para a execução da cadeia de ataque, com a função primária de iniciar a sequência de execução que ocorre inteiramente na memória. Não foi possível determinar se a DLL foi instalada com privilégios de administrador ou de usuário comum. Utilizando a função **GetSystemFirmwareTable** da API do Windows, o carregador acessa a tabela SMBIOS, que desde o Windows 10 versão 1803, está disponível para qualquer aplicativo em modo usuário. O SMBIOS é o padrão para fornecer informações de gerenciamento do sistema via firmware.

A função GetSystemFirmwareTable recupera dados da tabela **SMBIOS**, que são usados como chave para descriptografar o carregador RollSling, criptografado com XOR. Sem a chave correta de 32 bytes, a descriptografia do RollSling falharia, impedindo o avanço do malware. Isso indica um ataque altamente específico. Antes de registrar o RollFling como serviço, o invasor precisava coletar dados da tabela SMBIOS e enviá-los ao servidor C&C, que poderia então enviar um estágio adicional de ataque. Este estágio, chamado RollSling, fica armazenado na mesma pasta que o RollFling, mas com a extensão “. nls". Com a descriptografia bem-sucedida via XOR, o RollFling está pronto para carregar o RollSling na memória e prosseguir com sua execução.

```
if ( GetModuleHandleExA(6u, ServiceMain, &phModule) )
{
  if ( GetModuleFileNameA(phModule, Filename, 0x104u) )
  {
    strcpy_s(nlsFile, 260uLL, Filename);
    strcat_s(nlsFile, 260uLL, ".nls");
    memset(v27, 0, 0x104uLL);
    hFile = CreateFileA(nlsFile, 0x80000000, 1u, 0LL, OPEN_EXISTING, 0x80u, 0LL);
    hFileNls = hFile;
    if ( hFile != -1LL )
    {
      GetFileSizeEx(hFile, &FileSize);
      mem = LocalAlloc(0x40u, FileSize.QuadPart);
      v3 = mem;
      if ( mem )
      {
        ReadFile(hFileNls, mem, FileSize.LowPart, &NumberOfBytesRead, 0LL);
        CloseHandle(hFileNls);
        numberOfBytesWrittentoTheBUffer = GetSystemFirmwareTable('RSMB', 0, 0LL, 0);
        FirmwareTable = LocalAlloc(0x40u, numberOfBytesWrittentoTheBUffer);
        GetSystemFirmwareTable('RSMB', 0, FirmwareTable, numberOfBytesWrittentoTheBUffer);
        SMBIOSTableData = FirmwareTable->SMBIOSTableData;
```

Figura 2 – Obtendo o provedor de tabela de firmware SMBIOS.

O loader RollSling, ativado pelo RollFling, opera diretamente na memória para se esquivar de softwares de segurança. Sua tarefa é identificar um binário blob, que contém múltiplos estágios do ataque e configurações, localizado na mesma pasta ou na Package Cache. Caso não esteja na pasta do RollSling, o carregador busca na Package Cache. Esse blob foi previamente inserido no sistema durante a infecção inicial. O binário blob é eficiente por conter todas as informações necessárias em um arquivo, e sua criptografia oferece uma camada adicional de ocultação. O RollSling verifica a pasta em busca do blob correto, lendo primeiramente 4 bytes para determinar o tamanho dos dados. Após a leitura, os dados são invertidos e submetidos a várias verificações, incluindo a do cabeçalho MZ. Se validado, o carregador procura pela função "StartAction" no binário. Se as condições forem satisfeitas, o próximo estágio, RollMid, é carregado na memória.

Antes de executar o RollMid, o malware cria duas pastas no diretório **ProgramData\\Package** Cache com nomes gerados aleatoriamente e extensão ".cab". O blob é movido para a primeira pasta, e um novo arquivo temporário é colocado na segunda. Os invasores utilizam a pasta Package Cache, normalmente destinada a arquivos de instalação, para camuflar os arquivos maliciosos. A extensão ".cab" é comum nessa pasta, o que ajuda a evitar detecção. Por fim, o carregador RollSling executa a função "StartAction" com argumentos específicos, incluindo os caminhos do loader RollFling, do blob e do arquivo temporário criado pelo RollMid.

```
fix_api();
DLL_folder = (char *)&FullyQualifiedPath_to_folder_where_is_module;
if ( FullyQualifiedPath_to_folder_where_is_module._Myres >= 0x10 )
  DLL_folder = FullyQualifiedPath_to_folder_where_is_module._Bx._Ptr;
FirstFile = FindFirstFileExA(DLL_folder, FindExInfoStandard, &FindFileData, FindExSearchNameMatch, 0LL, 0)
if ( FirstFile == (HANDLE)-1LL )
{
looking_in_another_path:
  load_binary_to_memory_and_execute_StartAction_export_function(0LL);
  if ( FirstFile == (HANDLE)-1LL )
    goto exit;
}
else
{
  while ( FindFileData.cFileName[0] == '.'
       && (!FindFileData.cFileName[1] || FindFileData.cFileName[1] == '.' && !FindFileData.cFileName[2])
       || (FindFileData.dwFileAttributes & 0x10) != 0
       || load_binary_to_memory_and_execute_StartAction_export_function(FindFileData.cFileName) )
  {
    if ( !FindNextFileA(FirstFile, &FindFileData) )
      goto looking_in_another_path;
  }
}
FindClose(FirstFile);
```

Figura 3 – Busca por um binário blob na mesma pasta do RollFling.

O papel do loader RollMid é essencial para ativar os componentes principais do ataque e as configurações armazenadas no blob, além de estabelecer a ligação com um servidor de comando e controle (C&C). Embora não tenhamos conseguido obter o blob, que é vital para entender completamente o ataque, recuperamos o loader RollMid e alguns binários na memória. O RollMid, situado no início do blob, é descrito pelos primeiros 4 bytes que indicam seu tamanho. Seguem-se dois outros binários e as configurações finais, todos sujeitos a compressão e criptografia AES para maior segurança.

Os primeiros 4 bytes, destacados em amarelo, são cruciais para a análise e transição para as próximas seções do blob. Após o RollMid, dois conjuntos de 4 bytes, um em amarelo e outro em verde, indicam os tamanhos das seções FIRST_ENCRYPTED_DLL e SECOND_ENCRYPTED_DLL, respectivamente. O valor em verde também faz parte da chave AES de 16 bytes usada para descriptografar a FIRST_ENCRYPTED_DLL. Com esses tamanhos, podemos acessar as configurações no final do blob. Para iniciar o contato com o servidor de comando e controle (C&C), o malware precisa extrair os endereços iniciais C&C da seção CONFIGURATION_DATA. Uma vez descriptografados, esses endereços permitem que o malware se conecte à primeira camada do servidor C&C, utilizando a função GetHtmlFromUrl, que provavelmente realiza um pedido HTTP GET. Em resposta, o servidor envia um arquivo HTML com o endereço da segunda camada do servidor C&C. O malware, então, estabelece comunicação com essa segunda camada através da função importada GetImageFromUrl, que sugere a execução de um pedido GET para obter uma imagem.

Os atacantes utilizam técnicas de esteganografia para esconder informações essenciais na imagem, as quais são fundamentais para a próxima etapa do ataque. Infelizmente, não foi possível identificar quais dados críticos estão camuflados na imagem obtida da segunda camada do servidor C&C.

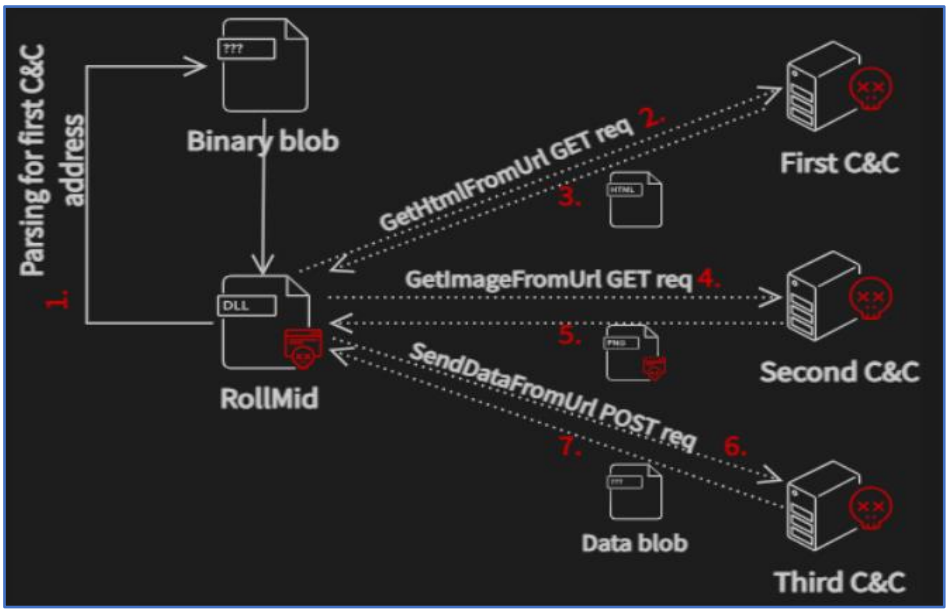


Figura 4 – Comunicação com servidores C&C..

Em uma etapa crítica do ataque, o uso de um Trojan de acesso remoto (RAT), especificamente o KaolinRAT, é essencial. Este RAT opera diretamente na memória e é ajustado com parâmetros particulares para sua funcionalidade. Ele vem completo com capacidades como a compressão de arquivos. No entanto, o KaolinRAT é apenas uma parte do ataque. Como discutido em um post anterior, existe um componente adicional, o rootkit FudModule.

Em uma telemetria avançada confirmou-se que o FudModule é implantado pelo KaolinRAT, evidenciando a integração e implantação eficazes entre eles. Isso destaca a complexidade e sofisticação da estratégia de ataque. Estabelecer uma comunicação segura com o servidor de comando e controle (C&C) do RAT é crucial, e isso é feito através da criptografia AES. Embora o binário responsável pela comunicação não esteja disponível, outros elementos da cadeia de ataque nos fornecem informações suficientes para fazer suposições educadas sobre o processo de comunicação.

O KaolinRAT é inicializado com seis argumentos, sendo um dos principais o endereço base do binário DLL do módulo de rede, que também foi utilizado no carregador RollMid. Outro argumento é os dados de configuração da segunda parte do blob de dados recebido. Para operar corretamente, o Kaolin RAT deve processar esses dados de configuração, que contêm informações como, intervalo de tempo de inatividade, indicador para coleta de dados sobre unidades de disco, indicador para recuperação de lista de sessões ativas de área de trabalho remota, endereços adicionais de servidores C&C.

Além disso, o KaolinRAT precisa carregar funções específicas do FIRST_DLL_BINARY, incluindo, **SendDataFromURL**, **ZipFolder**, **UnzipStr**, **curl_global_cleanup**, **curl_global_init**. Ainda que o método exato de envio de informações pelo KaolinRAT para o servidor C&C seja incerto, a presença de funções como "**curl_global_cleanup**" e "**curl_global_init**" indica que o envio provavelmente envolve chamadas de API da biblioteca curl.

Para iniciar a comunicação, o KaolinRAT envia uma solicitação POST ao servidor C&C. Nessa solicitação, ele constrói uma URL com o endereço do servidor C&C, um método semelhante ao do carregador RollMid. A URL é formada por:

"%addressOfC&Cserver%?%RandomWordFromDictonary%=%RandomString%"

O conteúdo da solicitação POST é criptografado com AES e codificado em base64. Dentro desse conteúdo, são incluídos dados como, o caminho de instalação do carregador RollFling e caminho para o blob, dados da chave de registro **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows\Iconservice**, KaolinID do processo RAT, nome do produto e número de compilação do sistema operacional, endereços de servidores C&C, nome do computador, diretório atual. Na solicitação POST com conteúdo criptografado, o malware anexa informações sobre a chave gerada e o vetor de inicialização, necessários para a descriptografia dos dados no backend. Isso é feito criando tuplas de valores-chave:

%RandomWordFromDictonary%=%TEMP_DATA%&%RandomWordFromDictonary%=%IV%%KEY%&%RandomWordFromDictonary%=%EncryptedContent%&%RandomWordFromDictonary%=%EncryptedHostNameAndIPAddr%

Com a comunicação estabelecida, o KaolinRAT está pronto para receber comandos. Os dados recebidos são criptografados e precisam ser descriptografados e analisados para executar um comando específico. Após processar o comando, o KaolinRAT envia os resultados de volta ao servidor C&C, também criptografados com a mesma chave AES e IV. Esses resultados podem incluir mensagens de erro, informações coletadas e o resultado das funções executadas.

O KaolinRAT pode executar vários comandos, como, atualizar o intervalo de inatividade, listar arquivos e coletar informações sobre discos disponíveis, atualizar, modificar ou excluir arquivos, alterar o carimbo de data/hora de um arquivo, listar processos ativos e seus módulos associados, criar ou encerrar processos, executar comandos via linha de comando, atualizar ou recuperar configurações internas, fazer upload de arquivos para o servidor C&C, conectar-se a um host arbitrário, compactar arquivos.

O RAT também pode baixar um arquivo DLL do servidor C&C e carregá-lo na memória, potencialmente executando funções exportadas como, **_DoMyFunc**, **_DoMyFunc2**, **_DoMyThread** (executa uma thread), **_DoMyCommandWork** e Configurar o diretório atual.

3 RECOMENDAÇÕES

Além dos indicadores de comprometimento elencados abaixo pela ISH, poderão ser adotadas medidas visando a mitigação da referida *ameaça*, como por exemplo:

Atualize regularmente

- Mantenha seus softwares antivírus, aplicativos e sistema operacional sempre atualizados.

Princípios de segurança de confiança zero

- Implemente princípios de segurança de confiança zero, concentrando suas estratégias de segurança em possíveis vetores de ataque de RAT.

Cuidado com phishing

- Esteja atento a e-mails suspeitos e não clique em links ou anexos desconhecidos.

Monitoramento de comportamento anormal

- Monitore aplicativos em busca de comportamentos anormais, como tráfego de rede inesperado.

Monitoramento de tráfego de rede

- Procure por tráfego de rede anômalo que possa estar associado a comunicações de RATs.

Aplicar patches

- Instale imediatamente os patches de segurança fornecidos pela Microsoft para esta vulnerabilidade específica.

Política de mínimo privilégio

- Implemente uma política de mínimo privilégio, permitindo aos usuários exercer ações na rede restritas ao desempenho de suas funções organizacionais.

4 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	
sha256:	a3fe80540363ee2f1216ec3d01209d7c517f6e749004c91901494fb94852332b
File name:	RollFling

Indicadores de compromisso do artefato	
sha256:	01ca7070bbe4bfa6254886f8599d6ce9537bafcbab6663f1f41bfc43f2ee370e
File name:	NLS files

Indicadores de compromisso do artefato	
sha256:	7248d66dea78a73b9b80b528d7e9f53bae7a77bad974edededeb16c33b14b9c56
File name:	NLS files

Indicadores de compromisso do artefato	
sha256:	e68ff1087c45a1711c3037dad427733ccb1211634d070b03cb3a3c7e836d210f
File name:	RollSling

Indicadores de compromisso do artefato	
sha256:	f47f78b5eef672e8e1bd0f26fb4aa699dec113d6225e2fcbd57129d6dada7def
File name:	RollSling

Indicadores de compromisso do artefato	
sha256:	9a4bc647c09775ed633c134643d18a0be8f37c21afa3c0f8adf41e038695643e
File name:	RollMid

Indicadores de compromisso do artefato	
sha256:	a75399f9492a8d2683d4406fa3e1320e84010b3affdff0b8f2444ac33ce3e690
File name:	Kaolin RAT

Indicadores de compromisso do artefato	
sha256:	b8a4c1792ce2ec15611932437a4a1a7e43b7c3783870afebf6eae043bcfade30
File name:	ISO

Tabela 1 – Indicadores de Compromissos de artefatos

Indicadores de URL, IPs e Domínios

Indicadores de URL, IPs e Domínios	
URL	https://www.henraux.com/ https://www.henraux.com/sitemaps/about/about.asp
IP	193.70.64.172 104.21.86.40

Tabela 2 – 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.

5 REFERÊNCIAS

- **Heimdall *by* ISH Tecnologia**
- Avast
- Thehackernews
- NVD

heimdall
security research
A DIVISION OF ISH