securelist.com /revengehotels-attacks-with-ai-and-venomrat-across-latin-america/117493/

RevengeHotels: a new wave of attacks leveraging LLMs and VenomRAT

Lisandro Ubiedo ⋮ ⋮ 9/16/2025

Authors

- Expert Lisandro Ubiedo

Background

RevengeHotels, also known as TA558, is a threat group that has been active since 2015, stealing credit card data from hotel guests and travelers. RevengeHotels' modus operandi involves sending emails with phishing links which redirect victims to websites mimicking document storage. These sites, in turn, download script files to ultimately infect the targeted machines. The final payloads consist of various remote access Trojan (RAT) implants, which enable the threat actor to issue commands for controlling compromised systems, stealing sensitive data, and maintaining persistence, among other malicious activities.

In previous campaigns, the group was observed using malicious emails with Word, Excel, or PDF documents attached. Some of them exploited the CVE-2017-0199 vulnerability, loading Visual Basic Scripting (VBS), or PowerShell scripts to install customized versions of different RAT families, such as RevengeRAT, NanoCoreRAT, NjRAT, 888 RAT, and custom malware named ProCC. These campaigns affected hotels in

multiple countries across Latin America, including Brazil, Argentina, Chile, and Mexico, but also hotel front-desks globally, particularly in Russia, Belarus, Turkey, and so on.

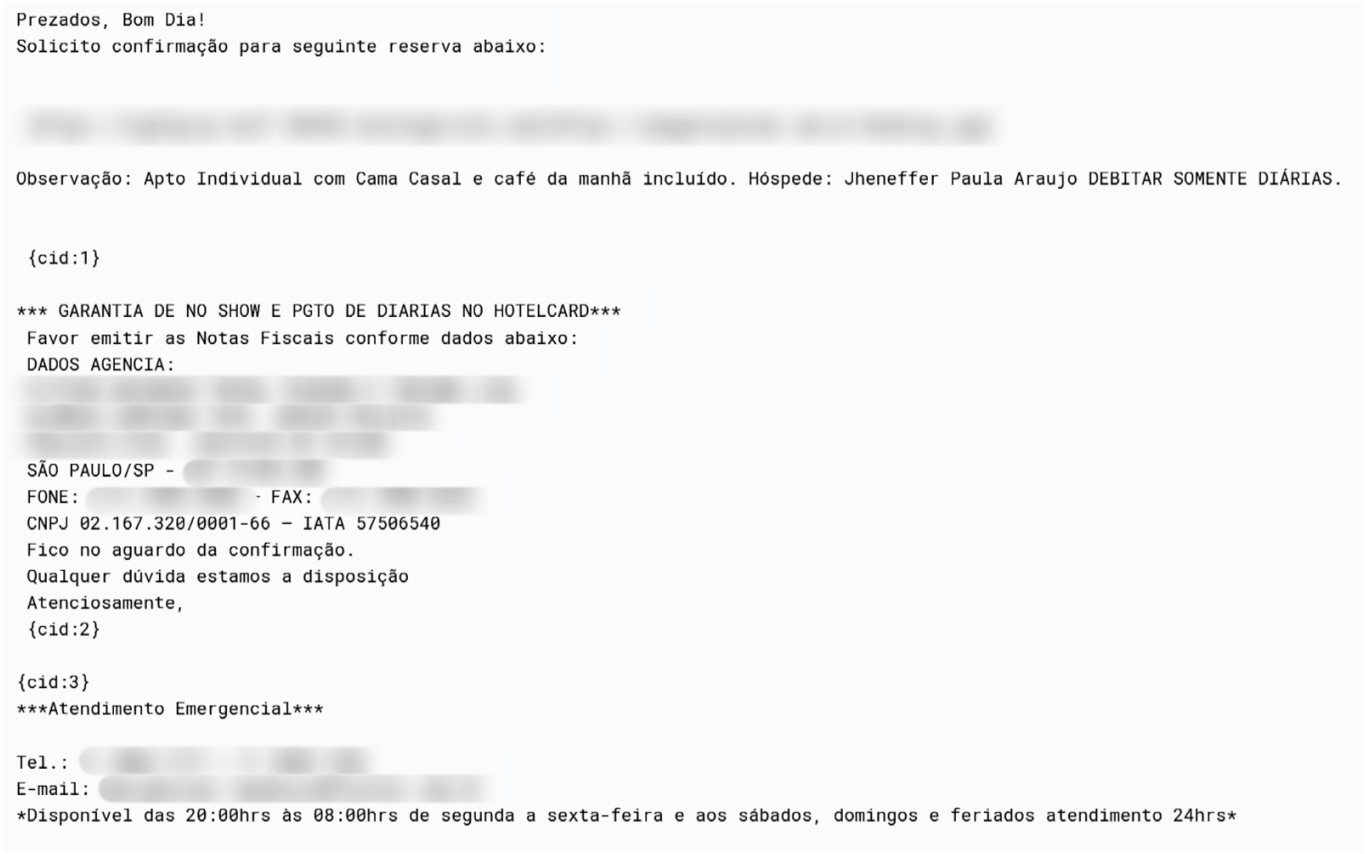
Later, this threat group expanded its arsenal by adding XWorm, a RAT with commands for control, data theft, and persistence, amongst other things. While investigating the campaign that distributed XWorm, we identified high-confidence indicators that RevengeHotels also used the RAT tool named DesckVBRAT in their operations.

In the summer of 2025, we observed new campaigns targeting the same sector and featuring increasingly sophisticated implants and tools. The threat actors continue to employ phishing emails with invoice themes to deliver VenomRAT implants via JavaScript loaders and PowerShell downloaders. A significant portion of the initial infector and downloader code in this campaign appears to be generated by large language model (LLM) agents. This suggests that the threat actor is now leveraging AI to evolve its capabilities, a trend also reported among other cybercriminal groups.

The primary targets of these campaigns are Brazilian hotels, although we have also observed attacks directed at Spanish-speaking markets. Through a comprehensive analysis of the attack patterns and the threat actor's modus operandi, we have established with high confidence that the responsible actor is indeed RevengeHotels. The consistency of the tactics, techniques, and procedures (TTPs) employed in these attacks aligns with the known behavior of RevengeHotels. The infrastructure used for payload delivery relies on legitimate hosting services, often utilizing Portuguese-themed domain names.

Initial infection

The primary attack vector employed by RevengeHotels is phishing emails with invoicing themes, which urge the recipient to settle overdue payments. These emails are specifically targeted at email addresses associated with hotel reservations. While Portuguese is a common language used in these phishing emails, we have also discovered instances of Spanish-language phishing emails, indicating that the threat actor's scope extends beyond Brazilian hospitality establishments and may include targets in Spanish-speaking countries or regions.



Prezados, Bom Dia!
Solicito confirmação para seguinte reserva abaixo:

Observação: Apto Individual com Cama Casal e café da manhã incluído. Hóspede: Jheneffer Paula Araujo DEBITAR SOMENTE DIÁRIAS.

{cid:1}

*** GARANTIA DE NO SHOW E PGTO DE DIARIAS NO HOTELCARD***
Favor emitir as Notas Fiscais conforme dados abaixo:
DADOS AGENCIA:
SÃO PAULO/SP -
FONE: · FAX:
CNPJ 02.167.320/0001-66 – IATA 57506540
Fico no aguardo da confirmação.
Qualquer dúvida estamos a disposição
Atenciosamente,
{cid:2}

{cid:3}
Atendimento Emergencial

Tel.:
E-mail:
Disponível das 20:00hrs às 08:00hrs de segunda a sexta-feira e aos sábados, domingos e feriados atendimento 24hrs

Example of a phishing email about a booking confirmation

In recent instances of these attacks, the themes have shifted from hotel reservations to fake job applications, where attackers sent résumés in an attempt to exploit potential job opportunities at the targeted hotels.

Malicious implant

The malicious websites, which change with each email, download a WScript JS file upon being visited, triggering the infection process. The filename of the JS file changes with every request. In the case at hand, we analyzed `Fat146571.js` (fbadfff7b61d820e3632a2f464079e8c), which follows the format `Fat\{NUMBER\}.js`, where “Fat” is the beginning of the Portuguese word “fatura”, meaning “invoice”.

The script appears to be generated by a large language model (LLM), as evidenced by its heavily commented code and a format similar to those produced by this type of technology. The primary function of the script is to load subsequent scripts that facilitate the infection.

A significant portion of the new generation of initial infectors created by RevengeHotels contains code that seems to have been generated by AI. These LLM-generated code segments can be distinguished from the original malicious code by several characteristics, including:

- The cleanliness and organization of the code
- Placeholders, which allow the threat actor to insert their own variables or content
- Detailed comments that accompany almost every action within the code

- A notable lack of obfuscation, which sets these LLM-generated sections apart from the rest of the code

AI Generated Code

```
    // Caminhos e configurações
    var directoryPath = "C:\\Users\\Public\\Scripts";
    var baseFileName = "SGDoHBZQWpLKXCAoTHXdBGlnQJLZCGBOVGLH";
    var extension = "ps1";
    var logFilePath = directoryPath + "\\operations.log";

    // Conteúdo do arquivo
    var content = reconstructed; // Defina 'reconstructed' anteriormente

    // Verificar ou criar diretório
    ensureDirectoryExists(directoryPath);

    // Gerar o nome completo do arquivo
    var fullFilePath = directoryPath + "\\" + generateFileName(baseFileName, extension);

    // Criar o arquivo
    createFile(fullFilePath, content);

    // Registrar a operação no log
    logOperation(logFilePath, "Arquivo criado: " + fullFilePath);

} catch (err) {

}
```

Marker by AI to use custom code

Custom Code

```
var jabiraca = "p&&&***&&&@@_*_@@er$$$$___$$$$hell"
jabiraca = jabiraca.replace ("&&&***&&&","o")
jabiraca = jabiraca.replace ("$$$$___$$$$","s")
jabiraca = jabiraca.replace ("@@_*_@@","w")


var HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY = "pow@#@$$$rsh@#@$$$ll.@#@$$$x@#@$$$ -@#@$$$x@#@$$
$cutionPolicy Bypas#########@@@ -Fil@#@$$$ \"" + fullFilePath  + "\"";
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY =
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY.replace ("@#@$$$","e")
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY =
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY.replace ("@#@$$$","e")
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY =
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY.replace ("@#@$$$","e")
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY =
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY.replace ("@#@$$$","e")
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY =
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY.replace ("@#@$$$","e")
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY =
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY.replace ("@#@$$$","e")
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY =
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY.replace ("@#@$$$","e")
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY =
HGHGHGHFGDGDTRFTYUYTYRTYTRYTYTYYRRRYTTYTYTYTYTYTY.replace ("#########@@@","s")
```

AI generated code in a malicious implant as compared to custom code

Second loading step

Upon execution, the loader script, `Fat\{NUMBER\}.js`, decodes an obfuscated and encoded buffer, which serves as the next step in loading the remaining malicious implants. This buffer is then saved to a PowerShell (PS1) file named `SGDoHBZQWpLKXCAoTHXdBGlnQJLZCGBOVGLH_{TIMESTAMP}.ps1`

(d5f241dee73cffe51897c15f36b713cc), where "\{TIMESTAMP\}" is a generated number based on the current execution date and time. This ensures that the filename changes with each infection and is not persistent. Once the script is saved, it is executed three times, after which the loader script exits.

The script `SGDoHBZQWpLKXCAoTHXdBGlnQJLZCGBOVGLH_{TIMESTAMP}.ps1` runs a PowerShell command with Base64-encoded code. This code retrieves the `cargajecerrr.txt` (b1a5dc66f40a38d807ec8350ae89d1e4) file from a remote malicious server and invokes it as PowerShell.

This downloader, which is lightly obfuscated, is responsible for fetching the remaining files from the malicious server and loading them. Both downloaded files are Base64-encoded and have descriptive names: `venumentrada.txt` (607f64b56bb3b94ee0009471f1fe9a3c), which can be interpreted as "VenomRAT entry point", and `runpe.txt` (dbf5afa377e3e761622e5f21af1f09e6), which is named after a malicious tool for in-memory execution. The first file, `venumentrada.txt`, is a heavily obfuscated loader (MD5 of the decoded file: 91454a68ca3a6ce7cb30c9264a88c0dc) that ensures the second file, a VenomRAT implant (3ac65326f598ee9930031c17ce158d3d), is correctly executed in memory.

The malicious code also exhibits characteristics consistent with generation by an AI interface, including a coherent code structure, detailed commenting, and explicit variable naming. Moreover, it differs significantly from previous samples, which had a structurally different, more obfuscated nature and lacked comments.

Exploring VenomRAT

VenomRAT, an evolution of the open-source QuasarRAT, was first discovered in mid-2020 and is offered on the dark web, with a lifetime license costing up to $650. Although the source code of VenomRAT was leaked, it is still being sold and used by threat actors.

VenomRAT packages on the dark web

According to the vendor's website, VenomRAT offers a range of capabilities that build upon and expand those of QuasarRAT, including HVNC hidden desktop, file grabber and stealer, reverse proxy, and UAC exploit, amongst others.

As with other RATs, VenomRAT clients are generated with custom configurations. The configuration data within the implant (similar to QuasarRAT) is encrypted using AES and PKCS #5 v2.0, with two keys employed: one for decrypting the data and another for verifying its authenticity using HMAC-SHA256. Throughout the malware code, different sets of keys and initialization vectors are used sporadically, but they consistently implement the same AES algorithm.

Anti-kill

It is notable that VenomRAT features an anti-kill protection mechanism, which can be enabled by the threat actor upon execution. Initially, the RAT calls a function named `EnableProtection`, which retrieves the security descriptor of the malicious process and modifies the Discretionary Access Control List (DACL) to remove any permissions that could hinder the RAT's proper functioning or shorten its lifespan on the system.

The second component of this anti-kill measure involves a thread that runs a continuous loop, checking the list of running processes every 50 milliseconds. The loop specifically targets those processes commonly used by security analysts and system administrators to monitor host activity or analyze .NET binaries, among other tasks. If the RAT detects any of these processes, it will terminate them without prompting the user.

```
string[] array = new string[107]
{
    "Taskmgr", "ProcessHacker", "cmd", "powershell", "regedit", "CCleaner", "Wireshark", "procmon64", "codecracker", "x96dbg",
    "pizza", "pepper", "reverse", "reversal", "de4dot", "pc-ret", "crack", "ILSpy", "x32dbg", "sharpod",
    "x64dbg", "x32_dbg", "x64_dbg", "debug", "dbg", "strongod", "PhantOm", "titanHide", "scyllaHide", "ilspy",
    "graywolf", "simpleassemblyexplorer", "MegaDumper", "megadumper", "X64NetDumper", "x64netdumper", "HxD", "hxd", "PETools", "petools",
    "Protection_ID", "protection_id", "die", "process hacker 2", "process", "hacker", "ollydbg", "x32dbg", "x64dbg", "ida -",
    "charles", "dnspy", "simpleassembly", "peek", "httpanalyzer", "httpdebug", "fiddler", "wireshark", "proxifier", "mitmproxy",
    "process hacker", "process monitor", "process hacker 2", "system explorer", "systemexplorer", "systemexplorerservice", "WPE PRO", "ghidra", "folderchangesview", "pc-ret",
    "folder", "dump", "proxy", "de4dotmodded", "StringDecryptor", "Centos", "SAE", "monitor", "brute", "checker",
    "zed", "sniffer", "http", "debugger", "james", "exeinfope", "codecracker", "x32dbg", "x64dbg", "ollydbg",
    "ida -", "charles", "dnspy", "simpleassembly", "peek", "httpanalyzer", "httpdebug", "fiddler", "wireshark", "dbx",
    "mdbg", "gdb", "windbg", "dbgclr", "kdb", "kgdb", "mdb"
};
```

List of processes that the malware looks for to terminate

The anti-kill measure also involves persistence, which is achieved through two mechanisms written into a VBS file generated and executed by VenomRAT. These mechanisms ensure the malware's continued presence on the system:

1. Windows Registry: The script creates a new key under HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce, pointing to the executable path. This allows the malware to persist across user sessions.
2. Process: The script runs a loop that checks for the presence of the malware process in the process list. If it is not found, the script executes the malware again.

If the user who executed the malware has administrator privileges, the malware takes additional steps to ensure its persistence. It sets the `SeDebugPrivilege` token, enabling it to use the `RtlSetProcessIsCritical` function to mark itself as a critical system process. This makes the process "essential" to the system, allowing it to persist even when termination is attempted. However, when the administrator logs off or the computer is about to shut down, VenomRAT removes its critical mark to permit the system to proceed with these actions.

As a final measure to maintain persistence, the RAT calls the `SetThreadExecutionState` function with a set of flags that forces the display to remain on and the system to stay in a working state. This prevents the system from entering sleep mode.

Separately from the anti-kill methods, the malware also includes a protection mechanism against Windows Defender. In this case, the RAT actively searches for `MSASCui.exe` in the process list and terminates it. The malware then modifies the task scheduler and registry to disable Windows Defender globally, along with its various features.

Networking

VenomRAT employs a custom packet building and serialization mechanism for its networking connection to the C2 server. Each packet is tailored to a specific action taken by the RAT, with a dedicated packet handler for each action. The packets transmitted to the C2 server undergo a multi-step process:

1. The packet is first serialized to prepare it for transmission.
2. The serialized packet is then compressed using LZMA compression to reduce its size.
3. The compressed packet is encrypted using AES-128 encryption, utilizing the same key and authentication key mentioned earlier.

Upon receiving packets from the C2 server, VenomRAT reverses this process to decrypt and extract the contents.

Additionally, VenomRAT implements tunneling by installing ngrok on the infected computer. The C2 server specifies the token, protocol, and port for the tunnel, which are sent in the serialized packet. This allows remote control services like RDP and VNC to operate through the tunnel and to be exposed to the internet.

USB spreading

VenomRAT also possesses the capability to spread via USB drives. To achieve this, it scans drive letters from C to M and checks if each drive is removable. If a removable drive is detected, the RAT copies itself to all available drives under the name `My Pictures.exe`.

Extra stealth steps

In addition to copying itself to another directory and changing its executable name, VenomRAT employs several stealth techniques that distinguish it from QuasarRAT. Two notable examples include:

- Deletion of Zone.Identifier streams: VenomRAT deletes the Mark of the Web streams, which contain metadata about the URL from which the executable was downloaded. By removing this information, the RAT can evade detection by security tools like Windows Defender and avoid being quarantined, while also eliminating its digital footprint.
- Clearing Windows event logs: The malware clears all Windows event logs on the compromised system, effectively creating a “clean slate” for its operations. This action ensures that any events generated during the RAT’s execution are erased, making it more challenging for security analysts to detect and track its activities.

Victimology

The primary targets of RevengeHotels attacks continue to be hotels and front desks, with a focus on establishments located in Brazil. However, the threat actors have been adapting their tactics, and phishing emails are now being sent in languages other than Portuguese. Specifically, we’ve observed that emails in Spanish are being used to target hotels and tourism companies in Spanish-speaking countries, indicating a potential expansion of the threat actor’s scope. Note that among earlier victims of this threat are such Spanish-speaking countries as Argentina, Bolivia, Chile, Costa Rica, Mexico, and Spain.

It is important to point out that previously reported campaigns have mentioned the threat actor targeting hotel front desks globally, particularly in Russia, Belarus, and Turkey, although no such activity has yet been detected during the latest RevengeHotels campaign.

Conclusions

RevengeHotels has significantly enhanced its capabilities, developing new tactics to target the hospitality and tourism sectors. With the assistance of LLM agents, the group has been able to generate and modify their phishing lures, expanding their attacks to new regions. The websites used for these attacks are constantly rotating, and the initial payloads are continually changing, but the ultimate objective remains the same: to deploy a remote access Trojan (RAT). In this case, the RAT in question is VenomRAT, a privately developed variant of the open-source QuasarRAT.

Kaspersky products detect these threats as `HEUR:Trojan-Downloader.Script.Agent.gen`, `HEUR:Trojan.Win32.Generic`, `HEUR:Trojan.MSIL.Agent.gen`, `Trojan-Downloader.PowerShell.Agent.ady`, `Trojan.PowerShell.Agent.aqx`.

Indicators of compromise

fbadfff7b61d820e3632a2f464079e8c Fat146571.js
d5f241dee73cffe51897c15f36b713cc
SGDoHBZQWpLKXCAoTHXdBGlnQJLZCGBOVGLH_{TIMESTAMP}.ps1
1077ea936033ee9e9bf444dafb55867c cargajecerrr.txt
b1a5dc66f40a38d807ec8350ae89d1e4 cargajecerrr.txt

dbf5afa377e3e761622e5f21af1f09e6 runpe.txt
607f64b56bb3b94ee0009471f1fe9a3c venumentrada.txt
3ac65326f598ee9930031c17ce158d3d deobfuscated runpe.txt
91454a68ca3a6ce7cb30c9264a88c0dc deobfuscated venumentrada.txt

- Malware Technologies
- Targeted attacks
- Malware Descriptions
- Malware
- RAT Trojan
- .NET
- PowerShell
- Brazil
- Data theft
- Thematic phishing
- LLM
- AI

Authors

- 

Lisandro Ubiedo

RevengeHotels: a new wave of attacks leveraging LLMs and VenomRAT

Your email address will not be published. Required fields are marked *

Cancel

This site uses Akismet to reduce spam. Learn how your comment data is processed.