

Analysis of Backdoor.WIN32.Buterat

Lat61 Threat Intelligence Team : : 9/9/2025



Introduction:

Backdoor malware is a covert type of malicious software designed to bypass standard authentication mechanisms and provide persistent, unauthorized access to compromised systems. Unlike conventional malware that prioritizes immediate damage or data theft, backdoors focus on stealth and longevity, enabling attackers to control infected endpoints remotely, deploy additional payloads, exfiltrate sensitive information, and move laterally across networks with minimal detection.

The Buterat backdoor is a notable example of this threat class, known for its sophisticated persistence techniques and adaptive communication methods with remote command-and-control (C2) servers. First identified in targeted attacks against enterprise and government networks, Buterat commonly spreads through phishing campaigns, malicious attachments, or trojanized software downloads. Once executed, it disguises its processes under legitimate system tasks, modifies registry keys for persistence, and uses encrypted or obfuscated communication channels to avoid network-based detection.

Preliminary Information About Sample:

MD5: 5d73aad06259533c238f0cdb3280d5a8

SHA-1: 6a1c418664fe5214c8e3d2f8f5020e1cb4311584

SHA-256: f50ec4cf0d0472a3e40ff8b9d713fb0995e648ecedf15082a88b6e6f1789cdab

Imphash: 3a1c6ade174e0b7afaa15737bba99cab

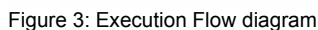
Compiler: Borland Delphi

Static Analysis:



Figure 2: Encrypted and Obfuscated strings

Dynamic Analysis:



00410AD8	55	push ebp	OptionalHeader.AddressOfEntryPoint
00410AD9	8BEC	mov ebp,esp	
00410ADB	B9 0B000000	mov ecx,B	ecx:EntryPoint, 0B:'\v'
00410AE0	6A 00	push 0	
00410AE2	6A 00	push 0	
00410AE4	49	dec ecx	ecx:EntryPoint
00410AE5	75 F9	jne 5d73aad06259533c238f0cdb3280d5a8.410AE0	
00410AE7	51	push ecx	ecx:EntryPoint
00410AE8	53	push ebx	
00410AE9	56	push esi	esi:EntryPoint
00410AEA	57	push edi	edi:EntryPoint
00410AEB	B8 800A4100	mov eax,5d73aad06259533c238f0cdb3280d5a8.410A80	
00410AF0	E8 3840FFFF	call 5d73aad06259533c238f0cdb3280d5a8.405830	
00410AF5	BE D86A4100	mov esi,5d73aad06259533c238f0cdb3280d5a8.416AD8	esi:EntryPoint
00410AFA	33C0	xor eax,eax	
00410AFC	55	push ebp	
00410AFD	68 BF3F4100	push 5d73aad06259533c238f0cdb3280d5a8.413F8F	
00410B02	64: FF30	push dword ptr [eax]	
00410B05	64: 8920	mov dword ptr [eax],101	
00410B08	68 01010000	push 101	
00410B0D	68 40684100	push 5d73aad06259533c238f0cdb3280d5a8.416840	
00410B12	6A 00	push 0	
00410B14	E8 434EFFFF	call <JMP.<GetModuleFileName>	
00410B19	B8 44694100	mov eax,5d73aad06259533c238f0cdb3280d5a8.416944	
00410B1E	BA 40684100	mov edx,5d73aad06259533c238f0cdb3280d5a8.416840	edx:EntryPoint
00410B23	B9 01010000	mov ecx,101	ecx:EntryPoint
00410B28	E8 4338FFFF	call 5d73aad06259533c238f0cdb3280d5a8.404370	
00410B2D	8D55 EC	lea edx,dword ptr ss:[ebp-14]	edx:EntryPoint
00410B30	A1 44694100	mov eax,dword ptr ds:[416944]	
00410B35	E8 BEFEFFFF	call 5d73aad06259533c238f0cdb3280d5a8.4109F8	
00410B3A	8B55 EC	mov edx,dword ptr ss:[ebp-14]	edx:EntryPoint
00410B3D	B8 48694100	mov eax,5d73aad06259533c238f0cdb3280d5a8.416948	
00410B42	E8 0D36FFFF	call 5d73aad06259533c238f0cdb3280d5a8.404154	
00410B47	68 E06A4100	push 5d73aad06259533c238f0cdb3280d5a8.416AE0	

Figure 4: Address of Entry Point

Entry point for the sample is at 0x00410AD8. So the user code execution starts at this address as shown in the above figure.

00413F28	A1 F4694100	mov eax,dword ptr ds:[4169F4]	
00413F2D	8B40 28	mov eax,dword ptr ds:[eax+28]	[eax+28]:EntryPoint
00413F30	0305 D06A4100	add eax,dword ptr ds:[416A00]	
00413F36	A1 A86A4100	mov dword ptr ds:[416A44],eax	
00413F3B	8D55 A8	lea edx,dword ptr ss:[ebp+58]	edx:EntryPoint
00413F3E	B8 80464100	mov eax,5d73aad06259533c238f0cdb3280d5a8.414680	414680: "S////////////////////////////////////e//t//t//h//r//e//a//d//C//o//n//t//e//x//t"
00413F43	E8 AC4AFFFF	call 5d73aad06259533c238f0cdb3280d5a8.40E3F4	
00413F48	8B45 A8	mov eax,dword ptr ss:[ebp+50]	
00413F4D	E8 4C06FFFF	call 5d73aad06259533c238f0cdb3280d5a8.40459C	
00413F50	50	push eax	
00413F51	A1 38684100	mov eax,dword ptr ds:[416838]	
00413F56	50	push eax	
00413F57	E8 101AFFFF	call <JMP.<GetProcAddress>	
00413F5C	A3 94694100	mov dword ptr ds:[416994],eax	
00413F61	68 F8694100	push 5d73aad06259533c238f0cdb3280d5a8.4169F8	
00413F66	A1 A4694100	mov eax,dword ptr ds:[4169A4]	
00413F6B	50	push eax	
00413F6C	F1F5 94694100	call dword ptr ds:[416994]	edx:EntryPoint
00413F72	8D55 A4	lea edx,dword ptr ss:[ebp+5C]	4146FC: "R////////////////////////////////////e//s//u//m//e//t//t//h//r//e//a//d//d"
00413F75	B8 FC464100	mov eax,5d73aad06259533c238f0cdb3280d5a8.4146FC	
00413F7A	E8 75A4FFFF	call 5d73aad06259533c238f0cdb3280d5a8.40E3F4	
00413F7F	8B45 A4	mov eax,dword ptr ss:[ebp+5C]	
00413F82	E8 1506FFFF	call 5d73aad06259533c238f0cdb3280d5a8.40459C	
00413F87	50	push eax	
00413F88	A1 38684100	mov eax,dword ptr ds:[416838]	
00413F8D	50	push eax	
00413F8E	E8 D919FFFF	call <JMP.<GetProcAddress>	
00413F93	A3 98694100	mov dword ptr ds:[416998],eax	
00413F98	A1 A4694100	mov eax,dword ptr ds:[4169A4]	

Figure 5: Obfuscated API calls

Some Obfuscated API calls:

- SetThreadContext

This API provides attackers with precise control over thread execution, enabling them to hijack existing threads without creating new ones or altering the process entry point. Such capabilities make it a preferred choice for stealthy payload delivery, thread injection, and evasion of lightweight behavioural detection mechanisms.

- ResumeThread

Resumes a thread whose info has been edited and changed by an attacker.

Files dropped during infection:

- C:\Users\Admin\amhost.exe
- C:\Users\Admin\bmhost.exe
- C:\Users\Admin\cmhost.exe
- C:\Users\Admin\dmhost.exe
- C:\Users\Admin\lqL1gG.exe

3D Objects	12/4/2024 10:46 PM	File folder	
Contacts	12/4/2024 10:46 PM	File folder	
Desktop	4/24/2025 12:59 PM	File folder	
Documents	12/4/2024 10:46 PM	File folder	
Downloads	12/4/2024 11:09 PM	File folder	
Favorites	12/4/2024 10:46 PM	File folder	
Links	12/4/2024 10:46 PM	File folder	
Music	12/4/2024 10:46 PM	File folder	
OneDrive	12/4/2024 10:48 PM	File folder	
Pictures	12/4/2024 10:48 PM	File folder	
Saved Games	12/4/2024 10:46 PM	File folder	
Searches	12/4/2024 10:47 PM	File folder	
Videos	12/4/2024 10:46 PM	File folder	
amhost.exe	8/11/2025 4:20 AM	Application	64 KB
bmhost.exe	8/11/2025 4:20 AM	Application	131 KB
cmhost.exe	8/11/2025 4:20 AM	Application	171 KB
dmhost.exe	8/11/2025 4:20 AM	Application	24 KB
lqL1gG.exe	8/11/2025 4:20 AM	Application	172 KB

Figure 6: Files Dropped during infection

Network Activity:

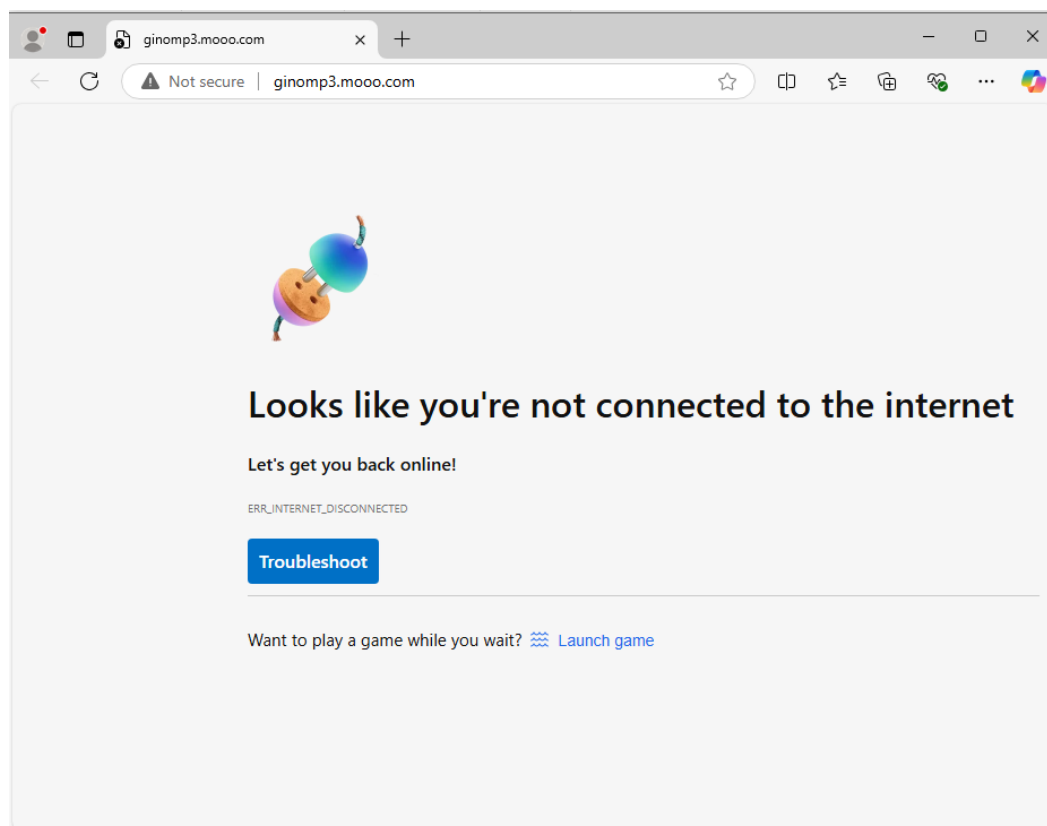


Figure 7: Sample trying to contact a suspicious link

Infection Prevention and Remediation:

What are the Technical Controls?

- Endpoint Protection:
 - Use up-to-date anti-malware and antivirus software to detect and remove threats like Backdoor.Buterat.
- Network Monitoring:
 - Implement tools and practices to monitor network traffic for suspicious connections. The malware connects to a remote C2 server at <http://ginomp3.mo00.com/>, which can be flagged as malicious activity.
- Firewall & IDS:

- Configure your firewall to block unauthorized access and use an Intrusion Detection System (IDS) to monitor for suspicious network activity.
- System Integrity Monitoring:
 - Monitor for unexpected file creation or modifications, such as the dropping of amhost.exe, bmhost.exe, cmhost.exe, dmhost.exe, and lqL1gG.exe in the C:\Users\Admin directory.
- Application Control:
 - Employ application control or allowlisting to prevent the execution of unauthorized or suspicious executables. This can stop the dropped files from running and further infecting the system.
- Obfuscation & Encryption Detection:
 - Be aware that the malware uses encrypted or obfuscated communication channels and hides its processes under legitimate system tasks to avoid detection. Use security solutions with behavioural analysis to identify unusual activity that may indicate the presence of this malware, even if its files are not detected by signature-based methods.

Employee Training & Awareness:

- Phishing campaigns: The Buterat backdoor commonly spreads through phishing campaigns and malicious attachments. Educate employees on how to identify and avoid suspicious emails and attachments.
- Malicious Downloads: The malware can also spread through trojanized software downloads. Users should only download software from trusted, official sources.

Incorporating Point Wild Product Protection:

Point Wild has a platform called Lat61 that unifies its security solutions, providing a framework for defense against threats like Backdoor.Buterat. The Lat61 Threat Intelligence Team at Point Wild actively tracks malware, its methods, infrastructure, and motivations.

Conclusion:

The Backdoor.Win32.Buterat malware demonstrates a highly stealthy and persistent infection methodology designed to maintain long-term unauthorized access to compromised systems. By leveraging encrypted strings, obfuscated API calls like SetThreadContext, and sophisticated thread manipulation techniques, it effectively bypasses standard behavioural detection mechanisms. Its ability to drop multiple payloads and establish communication with remote command-and-control (C2) infrastructure further amplifies its threat potential, enabling attackers to execute arbitrary commands, exfiltrate data, and deploy additional malicious components.

Given its advanced persistence and evasion tactics, timely detection and remediation are critical to preventing prolonged compromise. Security teams should implement network monitoring for unusual outbound traffic, file integrity checks, and memory analysis to identify injected threads and unauthorized processes. Proactive threat hunting, combined with updated signatures and behavioural detection rules, is essential to minimize exposure to Buterat and similar backdoor threats.

What are the Indicators of Compromise (IOCs)?

Hashes:

f50ec4cf0d0472a3e40ff8b9d713fb0995e648ecedf15082a88b6e6f1789cdab
 c5cdb87162f6e9162a92b461909a3624547e0ec8b9ffe36c2150ec5f78ce1164
 6d57ce74d2af2192a533127c658cad041a75f3210c9c9c66c27f9822c0a0b091
 2d72cf5e200110f5d75669441abe053be11d11768038f68860706dff54fc6be6
 73999153156cf5707c2cb88bea6c577a7abdc7ba2b62369fdb03cdc26f42e963
 86f5a5bb4153ae4b4b2129f112534638cb15527c28b0a771e4fd8ba8870daf77

Network:

Command and Control (C2) Server: <http://ginomp3.mo00.com/>