Frebniis: New Malware Abuses Microsoft IIS Feature to Establish Backdoor

Malware injects malicious code into Failed Request Event Buffering module in order to monitor HTTP requests from attacker.

Symantec, by Broadcom Software, has observed a new malware that abuses a feature of Microsoft's Internet Information Services (IIS) to deploy a backdoor onto targeted systems.

The malware, dubbed Frebniis (Backdoor.Frebniis), was used by a currently unknown threat actor against targets in Taiwan.

The technique used by Frebniis involves injecting malicious code into the memory of a DLL file (iisfreb.dll) related to an IIS feature used to troubleshoot and analyze failed web page requests. This allows the malware to stealthily monitor all HTTP requests and recognize specially formatted HTTP requests sent by the attacker, allowing for remote code execution. In order to use this technique, an attacker needs to gain access to the Windows system running the IIS server by some other means. In this particular case, it is unclear how this access was achieved.

Failed Request Event Buffering

IIS is a general-purpose web server that runs on Windows systems to serve requested HTML pages or files. An IIS web server accepts requests from remote client computers and returns the appropriate response. IIS

has a feature known as Failed Request Event Buffering (FREB) that collects data and details about requests, such as originating IP address and port, HTTP headers with cookies, etc.

A feature called Failed Request Tracing can be used to troubleshoot IIS failed requests. Failed Request Tracing buffers the trace events for a request and flushes them to disk if the request meets the definition of “fail” set by the user. Failed request tracing can, for example, be used to learn why requests are returning a specific HTTP status code (401 or 404, etc.), or why a request is taking too long to process, or is not responding.

Stealth Code Hijacking of IIS

Frebniis ensures Failed Request Tracing is enabled and then accesses w3wp.exe (IIS) process memory, obtaining the address of where the Failed Request Event Buffering code (iisfreb.dll) is loaded. With this code start address, Frebniis searches from there for a function pointer table to hijack code execution.

The authors of Frebniis have determined that a particular function pointer within iisfreb.dll is called by iiscore.dll whenever any HTTP request is made to IIS from a web client. This function normally checks if the content of the HTTP request matches the Failed Request Tracing rules.

```
.text:0000000180001320                  db 'ion.  Please check the
.text:0000000180001320                  db 'he corresponding metho
.text:00000001800013BC                  align 20h
.text:00000001800013C0 off_1800013C0    dq offset sub_1800048A0
.text:00000001800013C8                  dq offset sub_1800048A0
.text:00000001800013D0                  dq offset sub_1800048A0
.text:00000001800013D8                  dq offset sub_1800048A0
.text:00000001800013E0                  dq offset sub_1800048A0
```

Figure 1. Function pointer table used to hijack execution

Frebniis hijacks this function by injecting its own malicious code into IIS process memory and then replacing this function pointer with the address of its own malicious code. This hijack point allows Frebniis to stealthily receive and inspect every HTTP request to the IIS server before returning to the original function.

```
.text:00000001800011F8                  mov     rbx, [rsp+48h+arg_0]
.text:00000001800011FD                  mov     rsi, [rsp+48h+arg_8]
.text:0000000180001202                  add     rsp, 40h
.text:0000000180001206                  pop     rdi
.text:0000000180001207                  jmp     cs:qword_180025178
.text:0000000180001207 sub_18000119A0   endp
.text:0000000180001207
```

Figure 2. After running its own malicious code, Frebniis jumps back to the original function

Backdoor

The Frebniis malicious injected code parses all received HTTP POST requests for /logon.aspx or /default.aspx along with a parameter password set to ‘7ux4398!’. If the password matches, Frebniis decrypts and executes a section of the injected code, which is .NET executable code consisting of the main backdoor functionality. No executables are saved to disk in this process, keeping the backdoor completely stealthy.

The .NET code provides proxying functionality and remote code execution controlled by a provided second HTTP parameter that is a Base64 encoded string.

To enable the proxy, the encoded string is Base64 decoded and then decrypted (xor 0x08), with the first character representing a proxy command followed by expected parameters. The proxy is used to send and receive Base64 encoded data from other computer systems. This allows the attackers to communicate with internal resources that may normally be blocked from the internet via the compromised IIS server.

Table 1. Frebniis commands – the function names have been misspelled by the malware author

Command	Function name	Parameter	Description
1	CreateConnect	Host:Port	Connect to a remote system for proxying, returns a UUID representing the remote system
2	ReadScoket	Uuid	Read a Base64 string from a remote system
3	Writescoket	Uuid, Base64 string	Write a Base64 string to a remote system
4	CloseScoket	Uuid	Close the connection

The .NET backdoor code also supports remote execution. If an HTTP call to logon.aspx or default.aspx is received without the password parameter, but with the Base64 string, the Base64 string is assumed to be C# code that will be executed straight in memory. The Base64 string is decoded and then decrypted (xor 0x08) and is expected to be an XML document with the C# code to be executed in the '/doc' node under the 'data' attribute (E.g. <doc data=C# code>). The C# code is extracted and executed. This allows Frebniis to stealthily execute arbitrary code on the system.

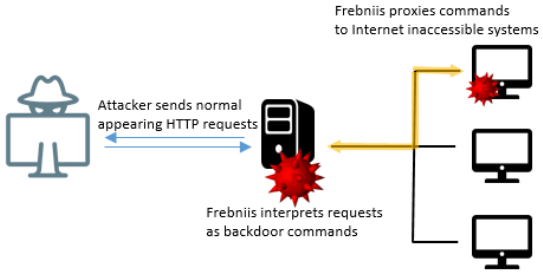

Figure 3. Example of how Frebniis is used

By hijacking and modifying IIS web server code, Frebniis is able to intercept the regular flow of HTTP request handling and look for specially formatted HTTP requests. These requests allow remote code execution and proxying to internal systems in a stealthy manner. No files or suspicious processes will be running on the system, making Frebniis a relatively unique and rare type of HTTP backdoor seen in the wild.

Protection/Mitigation

For the latest protection updates, please visit the Symantec Protection Bulletin.

Indicators of Compromise

If an IOC is malicious and the file available to us, Symantec Endpoint products will detect and block that file.

6464f9a5da26aa53fb2221255e908fd4da8edf0633f94051beee74a14b9b001c – Backdoor.Frebniis

b81c177c440e84635f22dc97b0411de93a24a983a41af676ffbbb4439487aaef – Backdoor.Frebniis