Barracuda Email Security Gateway Appliance (ESG) Vulnerability

By Barracuda Networks

Published: 2025-03-11 · Archived: 2026-04-05 12:51:40 UTC

JANUARY 4th, 2024:

On 12/29/2023, version 0.66 of Spreadsheet::ParseExcel was published. This release fixes CVE-2023-7101.

https://metacpan.org/dist/Spreadsheet-ParseExcel/changes

For organizations utilizing Spreadsheet::ParseExcel in their own products or services, we recommend reviewing CVE-2023-7101 and upgrading to the latest version of Spreadsheet::ParseExcel.

DECEMBER 24th, 2023:

In our ongoing investigation, Barracuda has determined that a threat actor has utilized an Arbitrary Code Execution (ACE) vulnerability within a third party library, Spreadsheet::ParseExcel, to deploy a specially crafted Excel email attachment to target a limited number of ESG devices. Spreadsheet::ParseExcel is an open source library used by the Amavis virus scanner within the ESG appliance. Barracuda, working in collaboration with Mandiant, assesses this activity is attributable to continued operations of the China nexus actor tracked as UNC4841.

On December 21, 2023, Barracuda deployed a security update to all active ESGs to address the ACE vulnerability in Spreadsheet::ParseExcel. The security update has been automatically applied, requiring no action by the customer.

Following UNC4841's exploitation of the ACE vulnerability (CVE-2023-7102), Barracuda has observed new variants of SEASPY and SALTWATER malware deployed to a limited number of ESG devices. On December 22, 2023, Barracuda deployed a patch to remediate compromised ESG appliances which exhibited indicators of compromise related to the newly identified malware variants.

No action is required by customers at this time, and our investigation is ongoing.

Barracuda has filed CVE-2023-7102 in relation to Barracuda's use of Spreadsheet::ParseExcel which has been patched. In addition, in order to increase public awareness of the ACE vulnerability in Spreadsheet::ParseExcel, Barracuda has filed CVE-2023-7101. At the time of this update, there is no known patch or update available to remediate CVE-2023-7101 within the open source library. For organizations utilizing Spreadsheet::ParseExcel in their own products or services, we recommend reviewing CVE-2023-7101 and promptly taking necessary remediation measures.

To assist organizations with hunting activity related to this UNC4841 activity. Indicators of Compromise have been added to the IOC tables below.

Current Indicators of Compromise (IOCs)

Host IOCs

Malware	MD5 Hash	SHA256	File Name(s)	File T
CVE-2023-7102 XLS Document	2b172fe3329260611a9022e71acdebca	803cb5a7de1fe0067a9eeb220dfc24ca56f3f571a986180e146b6cf387855bdd	ads2.xls	xls

CVE-2023-7102 XLS Document	e7842edc7868c8c5cf0480dd98bcfe76	952c5f45d203d8f1a7532e5b59af8e3306b5c1c53a30624b6733e0176d8d1acd	don.xls	xls
CVE-2023-7102 XLS Document	e7842edc7868c8c5cf0480dd98bcfe76	952c5f45d203d8f1a7532e5b59af8e3306b5c1c53a30624b6733e0176d8d1acd	personalbudget.xls	xls
SEASPY	7b83e4bd880bb9d7904e8f553c2736e3	118fad9e1f03b8b1abe00529c61dc3edfda043b787c9084180d83535b4d177b7	wifi-service	x-execut
SALTWATER	d493aab1319f10c633f6d223da232a27	34494ecb02a1cccadda1c7693c45666e1fe3928cc83576f8f07380801b07d8ba	mod_tll.so	x-shared

Network IOCs

IP Address	**ASN**	**Location**
23.224.99.242	40065	US
23.224.99.243	40065	US
23.224.99.244	40065	US
23.224.99.245	40065	US
23.224.99.246	40065	US
23.225.35.234	40065	US
23.225.35.235	40065	US
23.225.35.236	40065	US
23.225.35.237	40065	US
23.225.35.238	40065	US
107.148.41.146	398823	US

AUGUST 29th, 2023:

Today, Mandiant published an updated blog post (https://www.mandiant.com/resources/blog/unc4841-post-barracuda-zero-day-remediation) which further analyzed the actions of the Chinese-nexus threat group tracked as UNC4841. As noted in the blog, Mandiant and Barracuda have not identified any newly compromised ESG appliances post release of a security patch on May 20, 2023.which remediated the zero day ESG vulnerability (CVE-2023-2868). Mandiant assesses a limited number of previously impacted victims that have not followed Barracuda's guidance to replace their impacted appliances may still face risk associated with this.

Barracuda continues to recommend that impacted customers replace their compromised appliance. Only a limited number of ESG appliances worldwide were compromised and impacted customers have been notified to replace the appliances. Barracuda is providing the replacement product to impacted customers at no cost. No other Barracuda product, including Barracuda's SaaS email solutions, were impacted by this vulnerability.

JULY 28th, 2023:

While our investigation is still ongoing, Barracuda in conjunction with Mandiant, analyzed the additional malware code named SUBMARINE by CISA in its report issued on July 28, 2023 (https://www.cisa.gov/news-

events/alerts/2023/07/28/cisa-releases-malware-analysis-reports-barracuda-backdoors). This additional malware was utilized by the threat actor in response to Barracuda's remediation actions in an attempt to create persistent access on customer ESG appliances. This malware appeared on a very small number of already compromised ESG appliances. Barracuda's recommendation is unchanged. Customers should discontinue use of the compromised ESG appliance and contact Barracuda support (support@barracuda.com) to obtain a new ESG virtual or hardware appliance.

JUNE 15th, 2023:

Barracuda ESG Appliance Vulnerability Status Update

While our investigation is still ongoing, Barracuda now has a more comprehensive understanding of the incident, including that exploitation occurred on a subset of compromised Barracuda Email Security Gateway (ESG) appliances by an aggressive and highly skilled actor conducting targeted activity which, as reported by Mandiant, has suspected links to China. Consistent with our previous updates, we are sharing additional technical details to support our customers and partners. We are also publishing additional indicators of compromise that organizations can leverage for their network defenses.

For more technical details on the Barracuda ESG Zero-Day Vulnerability (CVE-2023-2868), please read Mandiant's blog at https://www.mandiant.com/resources/blog/barracuda-esg-exploited-globally. Along with this blog post, Mandiant has produced detailed Hardening Recommendations to assist organizations with this event.

Attribution

Mandiant assessed with high confidence that the threat actor, identified as UNC4841, who exploited the ESG zero-day vulnerability conducted targeted information gathering activity from a subset of organizations in support of the People's Republic of China.

Our priority throughout this incident has been transparency around what we know as well as the actions we've taken. As discussed in our guidance released on May 31, 2023, and reiterated on June 6, 2023, we recommend immediate replacement of compromised ESG appliances, regardless of patch level.

JUNE 15th, 2023:

Current Indicators of Compromise (IOCs)

Network IOCs

IP Address	**ASN**	**Netblock**	**Location**
101.229.146.218	4812	China Telecom	CN
103.146.179.101	136933	Gigabitbank Global	HK
103.27.108.62	132883	Topway Global Limited	HK
103.77.192.13	10222	Multibyte Info Technology Limited	HK
103.77.192.88	10222	Multibyte Info Technology Limited	HK
103.93.78.142	61414	Edgenap Ltd	JP
104.156.229.226	20473	Choopa, LLC	US
104.223.20.222	8100	CloudVPS	US
107.148.149.156	399195	Pegtechinc-ap-04	US
107.148.219.227	54600	Peg Tech	US

107.148.219.53	54600	Peg Tech	US
107.148.219.54	54600	Peg Tech	US
107.148.219.55	54600	Peg Tech	US
107.148.223.196	54600	Peg Tech	US
107.173.62.158	20278	Nexeon Technologies	US
137.175.19.25	54600	Peg Tech	US
137.175.28.251	54600	Peg Tech	US
137.175.30.36	54600	Peg Tech	US
137.175.30.86	54600	Peg Tech	US
137.175.51.147	54600	Peg Tech	US
137.175.53.17	54600	Peg Tech	US
137.175.53.170	54600	Peg Tech	US
137.175.53.218	54600	Peg Tech	US
137.175.60.252	54600	Peg Tech	US
137.175.60.253	54600	Peg Tech	US
137.175.78.66	54600	Peg Tech	US
139.84.227.9	20473	Choopa, LLC	ZA
155.94.160.72	8100	CloudVPS	US
182.239.114.135	9231	China Mobile Hong Kong	HK
182.239.114.254	9231	China Mobile Hong Kong	HK
192.74.226.142	54600	Peg Tech	CN
192.74.254.229	54600	Peg Tech	US
198.2.254.219	54600	Peg Tech	US
198.2.254.220	54600	Peg Tech	US
198.2.254.221	54600	Peg Tech	US
198.2.254.222	54600	Peg Tech	US
198.2.254.223	54600	Peg Tech	US
199.247.23.80	20473	Choopa, LLC	DE
213.156.153.34	202422	G-Core Labs S.A.	US
216.238.112.82	20473	Choopa, LLC	BR
23.224.42.29	40065	Cnservers LLC	US
23.224.78.130	40065	Cnservers LLC	US

23.224.78.131	40065	Cnservers LLC	US
23.224.78.132	40065	Cnservers LLC	US
23.224.78.133	40065	Cnservers LLC	US
23.224.78.134	40065	Cnservers LLC	US
37.9.35.217	202422	G-Core Labs S.A.	US
38.54.113.205	138915	Kaopu Cloud HK Limited	MY
38.54.1.82	138915	Kaopu Cloud HK Limited	SG
38.60.254.165	174	Cogent Communications	US
45.63.76.67	20473	Choopa, LLC	US
52.23.241.105	14618	Amazon.com	US
64.176.4.234	20473	Choopa, LLC	US
64.176.7.59	20473	Choopa, LLC	US
23.224.99.246	33330,133131		US
23.225.35.236	33330,133131		US

Domain
bestfindthetruth.com
fessionalwork.com
gesturefavour.com
goldenunder.com
singamofing.com
singnode.com
togetheroffway.com
troublendsef.com

Endpoint IOCs

Filename	**Hash**	**Type**
0d67f50a0bf7a3a017784146ac41ada0	snapshot.tar	Payload Attachment
42722b7d04f58dcb8bd80fe41c7ea09e	11111.tar	Payload Attachment
5392fb400bd671d4b185fb35a9b23fd3	imgdata.jpg	Payload Attachment
ac4fb6d0bfc871be6f68bfa647fc0125	snapshot.tar	Payload Attachment
878cf1de91f3ae543fd290c31adcbda4	snapshot.tar	Payload Attachment
b601fce4181b275954e3f35b18996c92	install_reuse.tar	SALTWATER install

827d507aa3bde0ef903ca5dec60cdec8	mod_udp.so	SALTWATER variant
c56d7b86e59c5c737ee7537d7cf13df1	autoins	SALTWATER install
6f79ef58b354fd33824c96625590c244	intent_reuse	SALTWATER install
349ca242bc6d2652d84146f5f91c3dbb	intentbas	SALTWATER install
1fea55b7c9d13d822a64b2370d015da7	mod_udp.so	SALTWATER variant
64c690f175a2d2fe38d3d7c0d0ddbb6e	mod_udp.so	SALTWATER variant
4cd0f3219e98ac2e9021b06af70ed643	mod_udp.so	SALTWATER variant
3b93b524db66f8bb3df8279a141734bb	mod_rtf.so	SALTWATER variant
8fdf3b7dc6d88594b8b5173c1aa2bc82	mod_rft.so	SALTWATER Variant
4ec4ceda84c580054f191caa09916c68	mod_rft.so	SALTWATER variant
1b1830abaf95bd5a44aa3873df901f28	mod_rft.so	SALTWATER variant
4ca4f582418b2cc0626700511a6315c0	BarracudaMailService	SEASPY Variant
c528b6398c86f8bdcfa3f9de7837ebfe	update_v2.sh	SEASPY Install
2d841cb153bebcfdee5c54472b017af2	rc	SEASPY launcher
c979e8651c1f40d685be2f66e8c2c610	rc	SEASPY launcher
1c042d39ca093b0e7f1412453b132076	rc	SEASPY launcher
ba7af4f98d85e5847c08cf6cefdf35dc	rc	SEASPY launcher
82eaf69de710abdc5dea7cd5cb56cf04	BarracudaMailService	SEASPY Variant
e80a85250263d58cc1a1dc39d6cf3942	BarracudaMailService	SEASPY Variant
5d6cba7909980a7b424b133fbac634ac	BarracudaMailService	SEASPY Variant
1bbb32610599d70397adfdaf56109ff3	BarracudaMailService	SEASPY Variant
4b511567cfa8dbaa32e11baf3268f074	BarracudaMailService	SEASPY Variant
a08a99e5224e1baf569fda816c991045	BarracudaMailService	SEASPY Variant
19ebfe05040a8508467f9415c8378f32	BarracudaMailService	SEASPY Variant
831d41ba2a0036540536c2f884d089f9	sendscd	SEASPY Variant
db4c48921537d67635bb210a9cb5bb52	BarracudaMailService	SEASPY Variant
694cdb49879f1321abb4605adf634935	install_bvp74_auth.tar	SEASPY install
5fdee67c82f5480edfa54afc5a9dc834	install_bvp74_auth.tar	SEASPY install
8fc03800c1179a18fbd58d746596fa7d	update_version	SEASPY launcher
17696a438387248a12cc911fbae8620e	resize_risertab	SEASPY launcher
4c1c2db989e0e881232c7748593d291e	update_version	SEASPY launcher
3e3f72f99062255d6320d5e686f0e212	update_version	SEASPY launcher

7d7fd05b262342a9e8237ce14ec41c3b	update_version	SEASPY launcher
2e30520f8536a27dd59eabbcb8e3532a	update_version	SEASPY launcher
0245e7f9105253ecb30de301842e28e4	update_version	SEASPY launcher
0c227990210e7e9d704c165abd76ebe2	update_version	SEASPY launcher
c7a89a215e74104682880def469d4758	update_version	SEASPY launcher
1bc5212a856f028747c062b66c3a722a	update_version	SEASPY launcher
a45ca19435c2976a29300128dc410fd4	update_version	SEASPY launcher
132a342273cd469a34938044e8f62482	update_version	SEASPY launcher
23f4f604f1a05c4abf2ac02f976b746b	resize2fstab	SEASPY Variant
45b79949276c9cb9cf5dc72597dc1006	resize_reisertab	SEASPY Variant
bef722484288e24258dd33922b1a7148	resize2fstab	SEASPY Variant
0805b523120cc2da3f71e5606255d29c	resize_reisertab	SEASPY Variant
69ef9a9e8d0506d957248e983d22b0d5	resize2fstab	SEASPY Variant
3c20617f089fe5cc9ba12c43c6c072f5	resize2fstab	SEASPY Variant
76811232ede58de2faf6aca8395f8427	resize2fstab	SEASPY Variant
f6857841a255b3b4e4eded7a66438696	resize_reisertab	SEASPY Variant
2ccb9759800154de817bf779a52d48f8	install_helo.tar	SEASIDE Install
cd2813f0260d63ad5adf0446253c2172	mod_require_helo.lua	SEASIDE variant
177add288b289d43236d2dba33e65956	rverify	WHIRLPOOL VARIANT
87847445f9524671022d70f2a812728f	mod_content.lua	SKIPJACK
35cf6faf442d325961935f660e2ab5a0	mod_attachment.lua	SEASPRAY
ce67bb99bc1e26f6cb1f968bc1b1ec21	install_att_v2.tar	SEASPRAY install
e4e86c273a2b67a605f5d4686783e0cc	mknod	SKIPJACK Persistence
ad1dc51a66201689d442499f70b78dea	get_fs_info.pl	SKIPJACK Persistence
9033dc5bac76542b9b752064a56c6ee4	nfsd_stub.ko	SANDBAR
e52871d82de01b7e7f134c776703f696	rverify	WHIRLPOOL Variant
446f3d71591afa37bbd604e2e400ae8b	mknod	SEASPRAY Persistence
666da297066a2596cacb13b3da9572bf	mod_sender.lua	SEASPRAY
436587bad5e061a7e594f9971d89c468	saslautchd	WHIRLPOOL Variant
85c5b6c408e4bdb87da6764a75008adf	rverify	WHIRLPOOL Variant
407738e565b4e9dafb07b782ebcf46b0	test1.sh	Reverse shell cronjob
cb0f7f216e8965f40a724bc15db7510b	update_v35.sh	Bash Script

N/A - multiple version identified	1.sh	Bash Script
19e373b13297de1783cecf856dc48eb0	cl	proxy client
N/A	aacore.sh	reverse shell cronjob
N/A	appcheck.sh	reverse shell cronjob
881b7846f8384c12c7481b23011d8e45	update_v31.sh	Bash Script
f5ab04a920302931a8bd063f27b745cc	intent_helo	Bash Script
N/A	p	Named pipe used in reverse shell
N/A	p7	Named pipe used in reverse shell
N/A	t	Named pipe used in reverse shell
N/A	core.sh	Reverse shell cronjob
N/A	p1	Named pipe used in reverse shell
177add288b289d43236d2dba33e65956	pd	WHIRLPOOL Variant
N/A	b	Named pipe used in reverse shell
d098fe9674b6b4cb540699c5eb452cb5	test.sh	Reverse shell cronjob
N/A	ss	Named pipe used in reverse shell

Detection Rules

Yara

```
rule M_Hunting_Exploit_Archive_2
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for TAR archives with /tmp/ base64 encoded being part of filename of enclosed files"
        md5 = "0d67f50a0bf7a3a017784146ac41ada0"
    strings:
        $ustar = { 75 73 74 61 72 }
        $b64_tmp = "/tmp/" base64
    condition:
        filesize < 1MB and
        $ustar at 257 and
        for any i in (0 .. #ustar) : (
            $b64_tmp in (i * 512 .. i * 512 + 250)
```

```
        )
}
rule M_Hunting_Exploit_Archive_3
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for TAR archive with openssl base64 encoded being part of filename of enclosed files"
        md5 = "0d67f50a0bf7a3a017784146ac41ada0"
    strings:
        $ustar = { 75 73 74 61 72 }
        $b64_openssl = "openssl" base64
    condition:
        filesize < 1MB and
        $ustar at 257 and
        for any i in (0 .. #ustar) : (
            $b64_openssl in (i * 512 .. i * 512 + 250)
        )
}
rule M_Hunting_Exploit_Archive_CVE_2023_2868
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for TAR archive with single quote/backtick as start of filename of enclosed files. CVE-2023-2868"
        md5 = "0d67f50a0bf7a3a017784146ac41ada0"
    strings:
        $ustar = { 75 73 74 61 72 }
        $qb = "'`"
    condition:
        filesize < 1MB and
        $ustar at 257 and
```

```
        for any i in (0 .. #ustar) : (
            $qb at (@ustar[i] + 255)
        )
}
rule M_Hunting_Linux_SALTWATER_1
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in SALTWATER samples."
        md5 = "827d507aa3bde0ef903ca5dec60cdec8"
    strings:
        $s1 = { 71 75 69 74 0D 0A 00 00 00 33 8C 25 3D 9C 17 70 08 F9 0C 1A 41 71 55 36 1A 5C 4B 8D 29 7E 0D 78 }
        $s2 = { 00 8B D5 AD 93 B7 54 D5 00 33 8C 25 3D 9C 17 70 08 F9 0C 1A 41 71 55 36 1A 5C 4B 8D 29 7E 0D 78 }
        $s3 = { 71 75 69 74 0D 0A 00 00 00 12 8D 03 07 9C 17 92 08 F0 0C 9A 01 06 08 00 1A 0C 0B 8D 18 0A 0D 0A }
    condition:
        uint32(0) == 0x464c457f and any of them
}
rule M_Hunting_Linux_SALTWATER_2
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in SALTWATER samples."
        md5 = "827d507aa3bde0ef903ca5dec60cdec8"
    strings:
        $c1 = "TunnelArgs"
        $c2 = "DownloadChannel"
        $c3 = "UploadChannel"
        $c4 = "ProxyChannel"
        $c5 = "ShellChannel"
        $c6 = "MyWriteAll"
        $c7 = "MyReadAll"
```

```
        $c8 = "Connected2Vps"
        $c9 = "CheckRemoteIp"
        $c10 = "GetFileSize"
        $s1 = "[-] error: popen failed"
        $s2 = "/home/product/code/config/ssl_engine_cert.pem"
        $s3 = "libbindshell.so"
    condition:
        uint32(0) == 0x464c457f and (any of ($s*) or 4 of ($c*))
}
rule FE_Hunting_Linux_Funchook_FEBeta
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in Funchook library - https://github.com/kubo/funchook"
        md5 = "827d507aa3bde0ef903ca5dec60cdec8"
    strings:
        $f = "funchook_"
        $s1 = "Enter funchook_create()"
        $s2 = "Leave funchook_create() => %p"
        $s3 = "Enter funchook_prepare(%p, %p, %p)"
        $s4 = "Leave funchook_prepare(..., [%p->%p],...) => %d"
        $s5 = "Enter funchook_install(%p, 0x%x)"
        $s6 = "Leave funchook_install() => %d"
        $s7 = "Enter funchook_uninstall(%p, 0x%x)"
        $s8 = "Leave funchook_uninstall() => %d"
        $s9 = "Enter funchook_destroy(%p)"
        $s10 = "Leave funchook_destroy() => %d"
        $s11 = "Could not modify already-installed funchook handle."
        $s12 = "  change %s address from %p to %p"
        $s13 = "  link_map addr=%p, name=%s"
        $s14 = "  ELF type is neither ET_EXEC nor ET_DYN."
```

```
        $s15 = "  not a valid ELF module %s."
        $s16 = "Failed to protect memory %p (size=%"
        $s17 = "  protect memory %p (size=%"
        $s18 = "Failed to unprotect memory %p (size=%"
        $s19 = "  unprotect memory %p (size=%"
        $s20 = "Failed to unprotect page %p (size=%"
        $s21 = "  unprotect page %p (size=%"
        $s22 = "Failed to protect page %p (size=%"
        $s23 = "  protect page %p (size=%"
        $s24 = "Failed to deallocate page %p (size=%"
        $s25 = " deallocate page %p (size=%"
        $s26 = "  allocate page %p (size=%"
        $s27 = "  try to allocate %p but %p (size=%"
        $s28 = "  allocate page %p (size=%"
        $s29 = "Could not find a free region near %p"
        $s30 = "  -- Use address %p or %p for function %p"
    condition:
        uint32(0) == 0x464c457f and (#f > 5 or 4 of ($s*))
}
rule M_Hunting_Linux_SEASPY_1
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in SEASPY samples."
        md5 = "4ca4f582418b2cc0626700511a6315c0"
    strings:
        $s1 = "usage: ./BarracudaMailService <Network-Interface>. e.g.: ./BarracudaMailService eth0"
        $s2 = "NO port code"
        $s3 = "pcap_lookupnet: %s"
        $s4 = "Child process id:%d"
        $s5 = "[*]Success!"
```

```
        $s6 = "enter open tty shell..."
    condition:
        uint32(0) == 0x464c457f and all of ($s*)
}
//
// SEASIDE
//
rule M_Hunting_Lua_SEASIDE_1
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in SEASIDE samples."
        md5 = "cd2813f0260d63ad5adf0446253c2172"
    strings:
        $s1 = "function on_helo()"
        $s2 = "local bindex,eindex = string.find(helo,'.onion')"
        $s3 = "helosend = 'pd'..' '..helosend"
        $s4 = "os.execute(helosend)"
    condition:
        (filesize < 1MB) and all of ($s*)
}
rule M_Hunting_SKIPJACK_1
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in SKIPJACK installation script."
        md5 = "e4e86c273a2b67a605f5d4686783e0cc"
    strings:
        $str1 = "hdr:name() == 'Content-ID'" base64
        $str2 = "hdr:body() ~= nil" base64
        $str3 = "string.match(hdr:body(),\"^[%w%+/=\\r\\n]+$\")" base64
```

```
        $str4 = "openssl aes-256-cbc" base64
        $str5 = "mod_content.lua"
        $str6 = "#!/bin/sh"
    condition:
        all of them
}
rule M_Hunting_Lua_SKIPJACK_2
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in SKIPJACK samples."
        md5 = "87847445f9524671022d70f2a812728f"
    strings:
        $str1 = "hdr:name() == 'Content-ID'"
        $str2 = "hdr:body() ~= nil"
        $str3 = "string.match(hdr:body(),\"^[%w%+/=\\r\\n]+$\")"
        $str4 = "openssl aes-256-cbc"
        $str5 = "| base64 -d| sh 2>"
    condition:
        all of them
}
rule M_Hunting_Lua_SEASPRAY_1
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in SEASPRAY samples."
        md5 = "35cf6faf442d325961935f660e2ab5a0"
    strings:
        $str1 = "string.find(attachment:filename(),'obt075') ~= nil"
        $str2 = "os.execute('cp '..tostring(tmpfile)..' /tmp/'..attachment:filename())"
        $str3 = "os.execute('rverify'..' /tmp/'..attachment:filename())"
```

```
    condition:
        all of them
}
rule M_Hunting_Linux_WHIRLPOOL_1
{
    meta:
        author = "Mandiant"
        description = "Hunting rule looking for strings observed in WHIRLPOOL samples."
        md5 = "177add288b289d43236d2dba33e65956"
    strings:
        $s1 = "error -1 exit" fullword
        $s2 = "create socket error: %s(error: %d)\n" fullword
        $s3 = "connect error: %s(error: %d)\n" fullword
        $s4 = {C7 00 20 32 3E 26 66 C7 40 04 31 00}
        $c1 = "plain_connect" fullword
        $c2 = "ssl_connect" fullword
        $c3 = "SSLShell.c" fullword
    condition:
        filesize < 15MB and uint32(0) == 0x464c457f and (all of ($s*) or all of ($c*))
}
```

Snort/Suricata

alert tcp any any -> <ESG_IP> [25,587] (msg:"M_Backdoor_SEASPY_oXmp"; flags:S; dsize:>9; content:"oXmp"; offset:0; depth:4; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000000; rev:1;)

alert tcp any any -> <ESG_IP> [25,587] (msg:"M_Backdoor_SEASPY_TfuZ"; flags:S; dsize:>9; content:"TfuZ"; offset:0; depth:4; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000001; rev:1;)

Suricata >= 5.0.4

alert tcp any any -> <ESG_IP> [25,587] (msg:"M_Backdoor_SEASPY_1358"; flags:S; tcp.hdr; content:"|05 4e|"; offset:22; depth:2; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000002; rev:1;)

alert tcp any any -> <ESG_IP> [25,587] (msg:"M_Backdoor_SEASPY_58928"; flags:S; tcp.hdr; content:"|e6 30|"; offset:28; depth:2; byte_test:4,>,16777216,0,big,relative; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000003; rev:1;)

alert tcp any any -> <ESG_IP> [25,587] (msg:"M_Backdoor_SEASPY_58930"; flags:S; tcp.hdr; content:"|e6 32|"; offset:28; depth:2; byte_test:4,>,16777216,0,big,relative; byte_test:2,>,0,0,big,relative; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000004; rev:1;)

alert tcp any any -> <ESG_IP> [25,587] (msg:"M_Backdoor_SEASPY_60826"; flags:S; tcp.hdr; content:"|ed 9a|"; offset:28; depth:2; byte_test:4,>,16777216,0,big,relative; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000005; rev:1;)

alert tcp any any -> <ESG_IP> [25,587] (msg:"M_Backdoor_SEASPY_60828"; flags:S; tcp.hdr; content:"|ed 9c|"; offset:28; depth:2; byte_test:4,>,16777216,0,big,relative; byte_test:2,>,0,0,big,relative; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000006; rev:1;)

JUNE 6th, 2023 (Updated on JUNE 15th, 2023):

Action Notice: Compromised ESG appliances must be immediately replaced regardless of patch version level. Only a subset of ESG appliances have shown any known indicators of compromise, and are identified by a message in the appliance User Interface.

If you have not replaced your appliance after receiving notice of compromise in your UI, contact Barracuda support (support@barracuda.com).

Barracuda's ESG appliance remediation recommendation for compromised appliances continues to be replacement of the compromised ESG.

JUNE 1st, 2023:

Preliminary Summary of Key Findings

Document History

Version/Date	Notes
1.0: May 30, 2023	Initial Document
1.1 : June 1, 2023	Additional IOCs and rules included

Barracuda Networks' priorities throughout this incident have been transparency and to use this as an opportunity to strengthen our policies, practices, and technology to further protect against future attacks. Although our investigation is ongoing, the purpose of this document is to share preliminary findings, provide the known Indicators of Compromise (IOCs), and share YARA rules to aid our customers in their investigations, including with respect to their own environments.

Timeline

- On May 18, 2023, Barracuda was alerted to anomalous traffic originating from Barracuda Email Security Gateway (ESG) appliances.
- On May 18, 2023, Barracuda engaged Mandiant, leading global cyber security experts, to assist in the investigation.
- On May 19, 2023, Barracuda identified a vulnerability (CVE-2023-28681) in our Email Security Gateway appliance (ESG).
- On May 20, 2023, a security patch to remediate the vulnerability was applied to all ESG appliances worldwide.
- On May 21, 2023, a script was deployed to all impacted appliances to contain the incident and counter unauthorized access methods.
- A series of security patches are being deployed to all appliances in furtherance of our containment strategy.

Key Findings

While the investigation is still on-going, Barracuda has concluded the following:

- The vulnerability existed in a module which initially screens the attachments of incoming emails. No other Barracuda products, including our SaaS email security services, were subject to the vulnerability identified.
- Earliest identified evidence of exploitation of CVE-2023-2868 is currently October 2022.

- Barracuda identified that CVE-2023-2868 was utilized to obtain unauthorized access to a subset of ESG appliances.
- Malware was identified on a subset of appliances allowing for persistent backdoor access.
- Evidence of data exfiltration was identified on a subset of impacted appliances..

Users whose appliances we believe were impacted have been notified via the ESG user interface of actions to take. Barracuda has also reached out to these specific customers. Additional customers may be identified in the course of the investigation.

CVE-2023-2868

On May 19, 2023, Barracuda Networks identified a remote command injection vulnerability (CVE-2023-2868) present in the Barracuda Email Security Gateway (appliance form factor only) versions 5.1.3.001-9.2.0.006. The vulnerability stemmed from incomplete input validation of user supplied .tar files as it pertains to the names of the files contained within the archive. Consequently, a remote attacker could format file names in a particular manner that would result in remotely executing a system command through Perl's qx operator with the privileges of the Email Security Gateway product.

Barracuda's investigation to date has determined that a third party utilized the technique described above to gain unauthorized access to a subset of ESG appliances.

Malware

This section details the malware that has been identified to date, and to assist in tracking, codenames for the malware have been assigned.

SALTWATER

SALTWATER is a trojanized module for the Barracuda SMTP daemon (bsmtpd) that contains backdoor functionality. The capabilities of SALTWATER include the ability to upload or download arbitrary files, execute commands, as well as proxy and tunneling capabilities.

Identified at path: /home/product/code/firmware/current/lib/smtp/modules on a subset of ESG appliances.

The backdoor is implemented using hooks on the send, recv, close syscalls and amounts to five components, most of which are referred to as “Channels” within the binary. In addition to providing proxying capabilities, these components exhibit backdoor functionality. The five (5) channels can be seen in the list below.

- DownloadChannel
- UploadChannel
- ProxyChannel
- ShellChannel
- TunnelArgs

Mandiant is still analyzing SALTWATER to determine if it overlaps with any other known malware families.

Table 1 below provides the file metadata related to a SALTWATER variant.

Name	**SHA256**		
mod_udp.so	1c6cad0ed66cf8fd438974e1eac0bc6dd9119f84892930cb71cb56a5e985f0a4		
MD5		**File Type**	**Size (Bytes)**
827d507aa3bde0ef903ca5dec60cdec8		ELF x86	1,879,643

Table 1: SALTWATER variant metadata

SEASPY

SEASPY is an x64 ELF persistence backdoor that poses as a legitimate Barracuda Networks service and establishes itself as a PCAP filter, specifically monitoring traffic on port 25 (SMTP) and port 587. SEASPY contains backdoor functionality that is activated by a "magic packet".

Identified at path: /sbin/ on a subset of ESG appliances.

Mandiant analysis has identified code overlap between SEASPY and cd00r, a publicly available backdoor.

Table 2 below provides the file metadata related to a SEASPY variant.

Name	**SHA256**		
BarracudaMailService	3f26a13f023ad0dcd7f2aa4e7771bba74910ee227b4b36ff72edc5f07336f115		
MD5		**File Type**	**Size (Bytes)**
4ca4f582418b2cc0626700511a6315c0		ELF x64	2,924,217

Table 2: SEASPY variant metadata

SEASIDE

SEASIDE is a Lua based module for the Barracuda SMTP daemon (bsmtpd) that monitors SMTP HELO/EHLO commands to receive a command and control (C2) IP address and port which it passes as arguments to an external binary that establishes a reverse shell.

Table 3 below provides the file metadata related to a SEASIDE.

Name	**SHA256**		
mod_require_helo.lua	fa8996766ae347ddcbbd1818fe3a878272653601a347d76ea3d5dfc227cd0bc8		
MD5		**File Type**	**Size (Bytes)**
cd2813f0260d63ad5adf0446253c2172		Lua module	2,724

Table 3: SEASIDE metadata

Recommendations For Impacted Customers

1. Ensure your ESG appliance is receiving and applying updates, definitions, and security patches from Barracuda. Contact Barracuda support (support@barracuda.com) to validate if the appliance is up to date.
2. Discontinue the use of the compromised ESG appliance and contact Barracuda support (support@barracuda.com) to obtain a new ESG virtual or hardware appliance.
3. Rotate any applicable credentials connected to the ESG appliance:
   - Any connected LDAP/AD
   - Barracuda Cloud Control
   - FTP Server
   - SMB
   - Any private TLS certificates
4. Review your network logs for any of the IOCs listed below and any unknown IPs. Contact compliance@barracuda.com if any are identified.

To support customers in the investigations of their environments, we are providing a list of all endpoint and network indicators observed over the course of the investigation to date. We have also developed a series of YARA rules that can be found in the section below.

Endpoint IOCs

Table 4 lists the endpoint IOCs, including malware and utilities, attributed to attacker activity during the investigation.

	File Name	MD5 Hash	Type
1	appcheck.sh	N/A	Bash script
2	aacore.sh	N/A	Bash script
3	1.sh	N/A	Bash script
4	mod_udp.so	827d507aa3bde0ef903ca5dec60cdec8	SALTWATER Variant
5	intent	N/A	N/A
6	install_helo.tar	2ccb9759800154de817bf779a52d48f8	TAR Package
7	intent_helo	f5ab04a920302931a8bd063f27b745cc	Bash script
8	pd	177add288b289d43236d2dba33e65956	Reverse Shell
9	update_v31.sh	881b7846f8384c12c7481b23011d8e45	Bash script
10	mod_require_helo.lua	cd2813f0260d63ad5adf0446253c2172	SEASIDE
11	BarracudaMailService	82eaf69de710abdc5dea7cd5cb56cf04	SEASPY
12	BarracudaMailService	e80a85250263d58cc1a1dc39d6cf3942	SEASPY
13	BarracudaMailService	5d6cba7909980a7b424b133fbac634ac	SEASPY
14	BarracudaMailService	1bbb32610599d70397adfdaf56109ff3	SEASPY
15	BarracudaMailService	4b511567cfa8dbaa32e11baf3268f074	SEASPY
16	BarracudaMailService	a08a99e5224e1baf569fda816c991045	SEASPY
17	BarracudaMailService	19ebfe05040a8508467f9415c8378f32	SEASPY
18	mod_udp.so	1fea55b7c9d13d822a64b2370d015da7	SALTWATER Variant
19	mod_udp.so	64c690f175a2d2fe38d3d7c0d0ddbb6e	SALTWATER Variant
20	mod_udp.so	4cd0f3219e98ac2e9021b06af70ed643	SALTWATER Variant

Table 4: Endpoint IOCs

Network IOCs

Table 5 lists the network IOCs, including IP addresses and domain names, attributed to attacker activity during the investigation.

	Indicator	ASN	Location
1	xxl17z.dnslog.cn	N/A	N/A
2	mx01.bestfindthetruth.com	N/A	N/A
3	64.176.7.59	AS-CHOOPA	US

4	64.176.4.234	AS-CHOOPA	US
5	52.23.241.105	AMAZON-AES	US
6	23.224.42.5	CloudRadium L.L.C	US
7	192.74.254.229	PEG TECH INC	US
8	192.74.226.142	PEG TECH INC	US
9	155.94.160.72	QuadraNet Enterprises LLC	US
10	139.84.227.9	AS-CHOOPA	US
11	137.175.60.253	PEG TECH INC	US
12	137.175.53.170	PEG TECH INC	US
13	137.175.51.147	PEG TECH INC	US
14	137.175.30.36	PEG TECH INC	US
15	137.175.28.251	PEG TECH INC	US
16	137.175.19.25	PEG TECH INC	US
17	107.148.219.227	PEG TECH INC	US
18	107.148.219.55	PEG TECH INC	US
19	107.148.219.54	PEG TECH INC	US
20	107.148.219.53	PEG TECH INC	US
21	107.148.219.227	PEG TECH INC	US
22	107.148.149.156	PEG TECH INC	US
23	104.223.20.222	QuadraNet Enterprises LLC	US
24	103.93.78.142	EDGENAP LTD	JP
25	103.27.108.62	TOPWAY GLOBAL LIMITED	HK
26	137.175.30.86	PEGTECHINC	US
27	199.247.23.80	AS-CHOOPA	DE
28	38.54.1.82	KAOPU CLOUD HK LIMITED	SG
29	107.148.223.196	PEGTECHINC	US
30	23.224.42.29	CNSERVERS	US
31	137.175.53.17	PEGTECHINC	US
32	103.146.179.101	GIGABITBANK GLOBAL	HK

Table 5: Network IOCs

YARA Rules

CVE-2023-2868

The following three (3) YARA rules can be used to hunt for the malicious TAR file which exploits CVE-2023-2868:

```
rule M_Hunting_Exploit_Archive_2
{
    meta:
        description = "Looks for TAR archive with /tmp/ base64 encoded being part of filename of enclosed files"
        date_created = "2023-05-26"
        date_modified = "2023-05-26"
        md5 = "0d67f50a0bf7a3a017784146ac41ada0"
        version = "1.0"
    strings:
        $ustar = { 75 73 74 61 72 }
        $b64_tmp = "/tmp/" base64
    condition:
        filesize < 1MB and

        $ustar at 257 and

        for any i in (0 .. #ustar) : (

            $b64_tmp in (i * 512 .. i * 512 + 250)

        )
}

rule M_Hunting_Exploit_Archive_3
{
    meta:
        description = "Looks for TAR archive with openssl base64 encoded being part of filename of enclosed files"
        date_created = "2023-05-26"
        date_modified = "2023-05-26"
        md5 = "0d67f50a0bf7a3a017784146ac41ada0"
        version = "1.0"
    strings:
        $ustar = { 75 73 74 61 72 }
        $b64_openssl = "openssl" base64
    condition:

        filesize < 1MB and
        $ustar at 257 and

        for any i in (0 .. #ustar) : (

            $b64_openssl in (i * 512 .. i * 512 + 250)

        )
}

rule M_Hunting_Exploit_Archive_CVE_2023_2868
{
    meta:
        description = "Looks for TAR archive with single quote/backtick as start of filename of enclosed files. CVE-2023-
2868"
        date_created = "2023-05-26"
```

```
        date_modified = "2023-05-26"
        md5 = "0d67f50a0bf7a3a017784146ac41ada0"
        version = "1.0"
    strings:
        $ustar = { 75 73 74 61 72 }
        $qb = "`"
    condition:

        filesize < 1MB and
        $ustar at 257 and

        for any i in (0 .. #ustar) : (

            $qb at (@ustar[i] + 255)

        )
}
```

SALTWATER

The following three (3) YARA rule can be used to hunt for SALTWATER:

```
rule M_Hunting_Linux_Funchook
{
    strings:
        $f = "funchook_"
        $s1 = "Enter funchook_create()"
        $s2 = "Leave funchook_create() => %p"
        $s3 = "Enter funchook_prepare(%p, %p, %p)"
        $s4 = "Leave funchook_prepare(..., [%p->%p],...) => %d"
        $s5 = "Enter funchook_install(%p, 0x%x)"
        $s6 = "Leave funchook_install() => %d"
        $s7 = "Enter funchook_uninstall(%p, 0x%x)"
        $s8 = "Leave funchook_uninstall() => %d"
        $s9 = "Enter funchook_destroy(%p)"
        $s10 = "Leave funchook_destroy() => %d"
        $s11 = "Could not modify already-installed funchook handle."
        $s12 = "  change %s address from %p to %p"
        $s13 = "  link_map addr=%p, name=%s"
        $s14 = "  ELF type is neither ET_EXEC nor ET_DYN."
        $s15 = "  not a valid ELF module %s."
        $s16 = "Failed to protect memory %p (size=%"
        $s17 = "  protect memory %p (size=%"
        $s18 = "Failed to unprotect memory %p (size=%"
        $s19 = "  unprotect memory %p (size=%"
        $s20 = "Failed to unprotect page %p (size=%"
        $s21 = "  unprotect page %p (size=%"
        $s22 = "Failed to protect page %p (size=%"
        $s23 = "  protect page %p (size=%"
        $s24 = "Failed to deallocate page %p (size=%"
        $s25 = " deallocate page %p (size=%"
        $s26 = "  allocate page %p (size=%"
        $s27 = "  try to allocate %p but %p (size=%"
```

```
        $s28 = "  allocate page %p (size=%"
        $s29 = "Could not find a free region near %p"
        $s30 = "  -- Use address %p or %p for function %p"
    condition:
        filesize < 15MB and uint32(0) == 0x464c457f and (#f > 5 or 4 of ($s*))
}

rule M_Hunting_Linux_SALTWATER_1
{
    strings:
        $s1 = { 71 75 69 74 0D 0A 00 00 00 33 8C 25 3D 9C 17 70 08 F9 0C 1A 41 71 55 36 1A 5C 4B 8D 29 7E 0D 78 }
        $s2 = { 00 8B D5 AD 93 B7 54 D5 00 33 8C 25 3D 9C 17 70 08 F9 0C 1A 41 71 55 36 1A 5C 4B 8D 29 7E 0D 78 }
    condition:
        filesize < 15MB and uint32(0) == 0x464c457f and any of them
}

rule M_Hunting_Linux_SALTWATER_2
{
    strings:
        $c1 = "TunnelArgs"
        $c2 = "DownloadChannel"
        $c3 = "UploadChannel"
        $c4 = "ProxyChannel"
        $c5 = "ShellChannel"
        $c6 = "MyWriteAll"
        $c7 = "MyReadAll"
        $c8 = "Connected2Vps"
        $c9 = "CheckRemoteIp"
        $c10 = "GetFileSize"
        $s1 = "[-] error: popen failed"
        $s2 = "/home/product/code/config/ssl_engine_cert.pem"
        $s3 = "libbindshell.so"
    condition:
        filesize < 15MB and uint32(0) == 0x464c457f and (2 of ($s*) or 4 of ($c*))
}
```

The following SNORT rule can be used to hunt for SEASPY magic packets:

alert tcp any any -> any [25,587] (msg:"M_Backdoor_SEASPY"; flags:S; dsize:>9; content:"oXmp"; offset:0; depth:4; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000000; rev:1;)

The following SNORT rules require Suricata 5.0.4 or newer and can be used to hunt for SEASPY magic packets:

alert tcp any any -> any [25,587] (msg:"M_Backdoor_SEASPY_1358"; flags:S; tcp.hdr; content:"|05 4e|"; offset:22; depth:2; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000001; rev:1;)

alert tcp any any -> any [25,587] (msg:"M_Backdoor_SEASPY_58928"; flags:S; tcp.hdr; content:"|e6 30|"; offset:28; depth:2; byte_test:4,>,16777216,0,big,relative; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000002; rev:1;)

alert tcp any any -> any [25,587] (msg:"M_Backdoor_SEASPY_58930"; flags:S; tcp.hdr; content:"|e6 32|"; offset:28; depth:2; byte_test:4,>,16777216,0,big,relative; byte_test:2,>,0,0,big,relative; threshold:type limit,track by_src,count 1,seconds 3600; sid:1000003; rev:1;)

MAY 30th, 2023:

Preliminary Summary of Key Findings

Barracuda Networks priorities throughout this incident have been transparency and to use this as an opportunity to strengthen our policies, practices, and technology to further protect against future attacks. Although our investigation is ongoing, the purpose of this document is to share preliminary findings, provide the known Indicators of Compromise (IOCs), and share YARA rules to aid our customers in their investigations, including with respect to their own environments.

Timeline

- On May 18, 2023, Barracuda was alerted to anomalous traffic originating from Barracuda Email Security Gateway (ESG) appliances.
- On May 18, 2023, Barracuda engaged Mandiant, leading global cyber security experts, to assist in the investigation.
- On May 19, 2023, Barracuda identified a vulnerability (CVE-2023-28681) in our Email Security Gateway appliance (ESG).
- On May 20, 2023, a security patch to remediate the vulnerability was applied to all ESG appliances worldwide.
- On May 21, 2023, a script was deployed to all impacted appliances to contain the incident and counter unauthorized access methods.
- A series of security patches are being deployed to all appliances in furtherance of our containment strategy.

Key Findings

While the investigation is still on-going, Barracuda has concluded the following:

- The vulnerability existed in a module which initially screens the attachments of incoming emails. No other Barracuda products, including our SaaS email security services, were subject to the vulnerability identified.
- Earliest identified evidence of exploitation of CVE-2023-2868 is currently October 2022.
- Barracuda identified that CVE-2023-2868 was utilized to obtain unauthorized access to a subset of ESG appliances.
- Malware was identified on a subset of appliances allowing for persistent backdoor access.
- Evidence of data exfiltration was identified on a subset of impacted appliances.

Users whose appliances we believe were impacted have been notified via the ESG user interface of actions to take. Barracuda has also reached out to these specific customers. Additional customers may be identified in the course of the investigation.

CVE-2023-2868

On May 19, 2023, Barracuda Networks identified a remote command injection vulnerability (CVE-2023-2868) present in the Barracuda Email Security Gateway (appliance form factor only) versions 5.1.3.001-9.2.0.006. The vulnerability stemmed from incomplete input validation of user supplied .tar files as it pertains to the names of the files contained within the archive. Consequently, a remote attacker could format file names in a particular manner that would result in remotely executing a system command through Perl's qx operator with the privileges of the Email Security Gateway product.

Barracuda's investigation to date has determined that a third party utilized the technique described above to gain unauthorized access to a subset of ESG appliances.

Malware

This section details the malware that has been identified to date.

SALTWATER

SALTWATER is a trojanized module for the Barracuda SMTP daemon (bsmtpd) that contains backdoor functionality. The capabilities of SALTWATER include the ability to upload or download arbitrary files, execute commands, as well as proxy and tunneling capabilities.

Identified at path: /home/product/code/firmware/current/lib/smtp/modules on a subset of ESG appliances.

The backdoor is implemented using hooks on the send, recv, close syscalls and amounts to five components, most of which are referred to as “Channels” within the binary. In addition to providing backdoor and proxying capabilities, these components exhibit classic backdoor functionality. The five (5) channels can be seen in the list below.

- DownloadChannel
- UploadChannel
- ProxyChannel
- ShellChannel
- TunnelArgs

Mandiant is still analyzing SALTWATER to determine if it overlaps with any other known malware families. Table 1 below provides the file metadata related to a SALTWATER variant.

Table 1 below provides the file metadata related to a SALTWATER variant.

Name	**SHA256**		
mod_udp.so	1c6cad0ed66cf8fd438974e1eac0bc6dd9119f84892930cb71cb56a5e985f0a4		
MD5		**File Type**	**Size (Bytes)**
827d507aa3bde0ef903ca5dec60cdec8		ELF x86	1,879,643

Table 1: SALTWATER variant metadata

SEASPY

SEASPY is an x64 ELF persistence backdoor that poses as a legitimate Barracuda Networks service and establishes itself as a PCAP filter, specifically monitoring traffic on port 25 (SMTP). SEASPY also contains backdoor functionality that is activated by a "magic packet".

Identified at path: /sbin/ on a subset of ESG appliances.

Mandiant analysis has identified code overlap between SEASPY and cd00r, a publicly available backdoor.

Table 2 below provides the file metadata related to a SEASPY variant.

Name	**SHA256**		
BarracudaMailService	3f26a13f023ad0dcd7f2aa4e7771bba74910ee227b4b36ff72edc5f07336f115		
MD5		**File Type**	**Size (Bytes)**
4ca4f582418b2cc0626700511a6315c0		ELF x64	2,924,217

Table 2: SEASPY variant metadata

SEASIDE

SEASIDE is a Lua based module for the Barracuda SMTP daemon (bsmtpd) that monitors SMTP HELO/EHLO commands to receive a command and control (C2) IP address and port which it passes as arguments to an external binary that establishes a reverse shell.

Table 3 below provides the file metadata related to a SEASIDE.

Name	**SHA256**

mod_require_helo.lua	fa8996766ae347ddcbbd1818fe3a878272653601a347d76ea3d5dfc227cd0bc8

MD5	**File Type**	**Size (Bytes)**
cd2813f0260d63ad5adf0446253c2172	Lua module	2,724

Table 3: SEASIDE metadata

Recommendations For Impacted Customers

1. Ensure your ESG appliance is receiving and applying updates, definitions, and security patches from Barracuda. Contact Barracuda support (support@barracuda.com) to validate if the appliance is up to date.
2. Discontinue the use of the compromised ESG appliance and contact Barracuda support (support@barracuda.com) to obtain a new ESG virtual or hardware appliance.
3. Rotate any applicable credentials connected to the ESG appliance:
   - Any connected LDAP/AD
   - Barracuda Cloud Control
   - FTP Server
   - SMB
   - Any private TLS certificates
4. Review your network logs for any of the IOCs listed below and any unknown IPs. Contact compliance@barracuda.com if any are identified.

To support customers in the investigations of their environments, we are providing a list of all endpoint and network indicators observed over the course of the investigation to date. We have also developed a series of YARA rules that can be found in the section below.

Endpoint IOCs

Table 4 lists the endpoint IOCs, including malware and utilities, attributed to attacker activity during the investigation.

	File Name	**MD5 Hash**	**Type**
1	appcheck.sh	N/A	Bash script
2	aacore.sh	N/A	Bash script
3	1.sh	N/A	Bash script
4	mod_udp.so	827d507aa3bde0ef903ca5dec60cdec8	SALTWATER Variant
5	intent	N/A	N/A
6	install_helo.tar	2ccb9759800154de817bf779a52d48f8	TAR Package
7	intent_helo	f5ab04a920302931a8bd063f27b745cc	Bash script
8	pd	177add288b289d43236d2dba33e65956	Reverse Shell
9	update_v31.sh	881b7846f8384c12c7481b23011d8e45	Bash script
10	mod_require_helo.lua	cd2813f0260d63ad5adf0446253c2172	SEASIDE
11	BarracudaMailService	82eaf69de710abdc5dea7cd5cb56cf04	SEASPY
12	BarracudaMailService	e80a85250263d58cc1a1dc39d6cf3942	SEASPY
13	BarracudaMailService	5d6cba7909980a7b424b133fbac634ac	SEASPY

14	BarracudaMailService	1bbb32610599d70397adfdaf56109ff3	SEASPY
15	BarracudaMailService	4b511567cfa8dbaa32e11baf3268f074	SEASPY
16	BarracudaMailService	a08a99e5224e1baf569fda816c991045	SEASPY
17	BarracudaMailService	19ebfe05040a8508467f9415c8378f32	SEASPY
18	mod_udp.so	1fea55b7c9d13d822a64b2370d015da7	SALTWATER Variant
19	mod_udp.so	64c690f175a2d2fe38d3d7c0d0ddbb6e	SALTWATER Variant
20	mod_udp.so	4cd0f3219e98ac2e9021b06af70ed643	SALTWATER Variant

Table 4: Endpoint IOCs

Network IOCs

Table 5 lists the network IOCs, including IP addresses and domain names, attributed to attacker activity during the investigation.

	Indicator	**ASN**	**Location**
1	xxl17z.dnslog.cn	N/A	N/A
2	mx01.bestfindthetruth.com	N/A	N/A
3	64.176.7.59	AS-CHOOPA	US
4	64.176.4.234	AS-CHOOPA	US
5	52.23.241.105	AMAZON-AES	US
6	23.224.42.5	CloudRadium L.L.C	US
7	192.74.254.229	PEG TECH INC	US
8	192.74.226.142	PEG TECH INC	US
9	155.94.160.72	QuadraNet Enterprises LLC	US
10	139.84.227.9	AS-CHOOPA	US
11	137.175.60.253	PEG TECH INC	US
12	137.175.53.170	PEG TECH INC	US
13	137.175.51.147	PEG TECH INC	US
14	137.175.30.36	PEG TECH INC	US
15	137.175.28.251	PEG TECH INC	US
16	137.175.19.25	PEG TECH INC	US
17	107.148.219.227	PEG TECH INC	US
18	107.148.219.55	PEG TECH INC	US
19	107.148.219.54	PEG TECH INC	US
20	107.148.219.53	PEG TECH INC	US

21	107.148.219.227	PEG TECH INC	US
22	107.148.149.156	PEG TECH INC	US
23	104.223.20.222	QuadraNet Enterprises LLC	US
24	103.93.78.142	EDGENAP LTD	JP
25	103.27.108.62	TOPWAY GLOBAL LIMITED	HK

Table 5: Network IOCs

YARA Rules

CVE-2023-2868

The following three (3) YARA rules can be used to hunt for the malicious TAR file which exploits CVE-2023-2868:

```
rule M_Hunting_Exploit_Archive_2
 {
    meta:
       description = "Looks for TAR archive with /tmp/ base64 encoded being part of filename of enclosed files"
       date_created = "2023-05-26"
       date_modified = "2023-05-26"
       md5 = "0d67f50a0bf7a3a017784146ac41ada0"
       version = "1.0"
    strings:
       $ustar = { 75 73 74 61 72 }
       $b64_tmp = "/tmp/" base64
    condition:
       filesize < 1MB and

       $ustar at 257 and

       for any i in (0 .. #ustar) : (

          $b64_tmp in (i * 512 .. i * 512 + 250)

       )
 }

rule M_Hunting_Exploit_Archive_3
 {
    meta:
       description = "Looks for TAR archive with openssl base64 encoded being part of filename of enclosed files"
       date_created = "2023-05-26"
       date_modified = "2023-05-26"
       md5 = "0d67f50a0bf7a3a017784146ac41ada0"
       version = "1.0"
    strings:
       $ustar = { 75 73 74 61 72 }
       $b64_openssl = "openssl" base64
    condition:

       filesize < 1MB and
       $ustar at 257 and
```

```
        for any i in (0 .. #ustar) : (

            $b64_openssl in (i * 512 .. i * 512 + 250)

        )
}

rule M_Hunting_Exploit_Archive_CVE_2023_2868
{
    meta:
        description = "Looks for TAR archive with single quote/backtick as start of filename of enclosed files. CVE-2023-2868"
        date_created = "2023-05-26"
        date_modified = "2023-05-26"
        md5 = "0d67f50a0bf7a3a017784146ac41ada0"
        version = "1.0"
    strings:
        $ustar = { 75 73 74 61 72 }
        $qb = "'`"
    condition:

        filesize < 1MB and
        $ustar at 257 and

        for any i in (0 .. #ustar) : (

            $qb at (@ustar[i] + 255)

        )
}
```

SALTWATER

The following three (3) YARA rule can be used to hunt for SALTWATER:

```
rule M_Hunting_Linux_Funchook
{
    strings:
        $f = "funchook_"
        $s1 = "Enter funchook_create()"
        $s2 = "Leave funchook_create() => %p"
        $s3 = "Enter funchook_prepare(%p, %p, %p)"
        $s4 = "Leave funchook_prepare(..., [%p->%p],...) => %d"
        $s5 = "Enter funchook_install(%p, 0x%x)"
        $s6 = "Leave funchook_install() => %d"
        $s7 = "Enter funchook_uninstall(%p, 0x%x)"
        $s8 = "Leave funchook_uninstall() => %d"
        $s9 = "Enter funchook_destroy(%p)"
        $s10 = "Leave funchook_destroy() => %d"
        $s11 = "Could not modify already-installed funchook handle."
        $s12 = "  change %s address from %p to %p"
        $s13 = "  link_map addr=%p, name=%s"
        $s14 = "  ELF type is neither ET_EXEC nor ET_DYN."
        $s15 = "  not a valid ELF module %s."
```

```
        $s16 = "Failed to protect memory %p (size=%"
        $s17 = "  protect memory %p (size=%"
        $s18 = "Failed to unprotect memory %p (size=%"
        $s19 = "  unprotect memory %p (size=%"
        $s20 = "Failed to unprotect page %p (size=%"
        $s21 = "  unprotect page %p (size=%"
        $s22 = "Failed to protect page %p (size=%"
        $s23 = "  protect page %p (size=%"
        $s24 = "Failed to deallocate page %p (size=%"
        $s25 = " deallocate page %p (size=%"
        $s26 = "  allocate page %p (size=%"
        $s27 = "  try to allocate %p but %p (size=%"
        $s28 = "  allocate page %p (size=%"
        $s29 = "Could not find a free region near %p"
        $s30 = "  -- Use address %p or %p for function %p"
    condition:
        filesize < 15MB and uint32(0) == 0x464c457f and (#f > 5 or 4 of ($s*))
}

rule M_Hunting_Linux_SALTWATER_1
{
    strings:
        $s1 = { 71 75 69 74 0D 0A 00 00 00 33 8C 25 3D 9C 17 70 08 F9 0C 1A 41 71 55 36 1A 5C 4B 8D 29 7E 0D 78 }
        $s2 = { 00 8B D5 AD 93 B7 54 D5 00 33 8C 25 3D 9C 17 70 08 F9 0C 1A 41 71 55 36 1A 5C 4B 8D 29 7E 0D 78 }
    condition:
        filesize < 15MB and uint32(0) == 0x464c457f and any of them
}

rule M_Hunting_Linux_SALTWATER_2
{
    strings:
        $c1 = "TunnelArgs"
        $c2 = "DownloadChannel"
        $c3 = "UploadChannel"
        $c4 = "ProxyChannel"
        $c5 = "ShellChannel"
        $c6 = "MyWriteAll"
        $c7 = "MyReadAll"
        $c8 = "Connected2Vps"
        $c9 = "CheckRemoteIp"
        $c10 = "GetFileSize"
        $s1 = "[-] error: popen failed"
        $s2 = "/home/product/code/config/ssl_engine_cert.pem"
        $s3 = "libbindshell.so"
    condition:
        filesize < 15MB and uint32(0) == 0x464c457f and (2 of ($s*) or 4 of ($c*))
}
```

MAY 23rd, 2023:

Barracuda identified a vulnerability (CVE-2023-2868) in our Email Security Gateway appliance (ESG) on May 19, 2023. A security patch to eliminate the vulnerability was applied to all ESG appliances worldwide on Saturday, May 20, 2023. The

vulnerability existed in a module which initially screens the attachments of incoming emails. No other Barracuda products, including our SaaS email security services, were subject to this vulnerability.

We took immediate steps to investigate this vulnerability. Based on our investigation to date, we've identified that the vulnerability resulted in unauthorized access to a subset of email gateway appliances. As part of our containment strategy, all ESG appliances have received a second patch on May 21, 2023. Users whose appliances we believe were impacted have been notified via the ESG user interface of actions to take. Barracuda has also reached out to these specific customers.

We will continue actively monitoring this situation, and we will be transparent in sharing details on what actions we are taking. Information gathering is ongoing as part of the investigation. We want to ensure we only share validated information with actionable steps for you to take. As we have information to share, we will provide updates via this product status page (https://status.barracuda.com) and direct outreach to impacted customers. Updates are also located on Barracuda's Trust Center (https://www.barracuda.com/company/legal).

Barracuda's investigation was limited to the ESG product, and not the customer's specific environment. Therefore, impacted customers should review their environments and determine any additional actions they want to take.

Your trust is important to us. We thank you for your understanding and support as we work through this issue and sincerely apologize for any inconvenience it may cause. If you have any questions, please reach out to support@barracuda.com.
