First Malware Targeting AWS Lambda Serverless Platform Discovered

thehackernews.com/2022/04/first-malware-targeting-aws-lambda.html

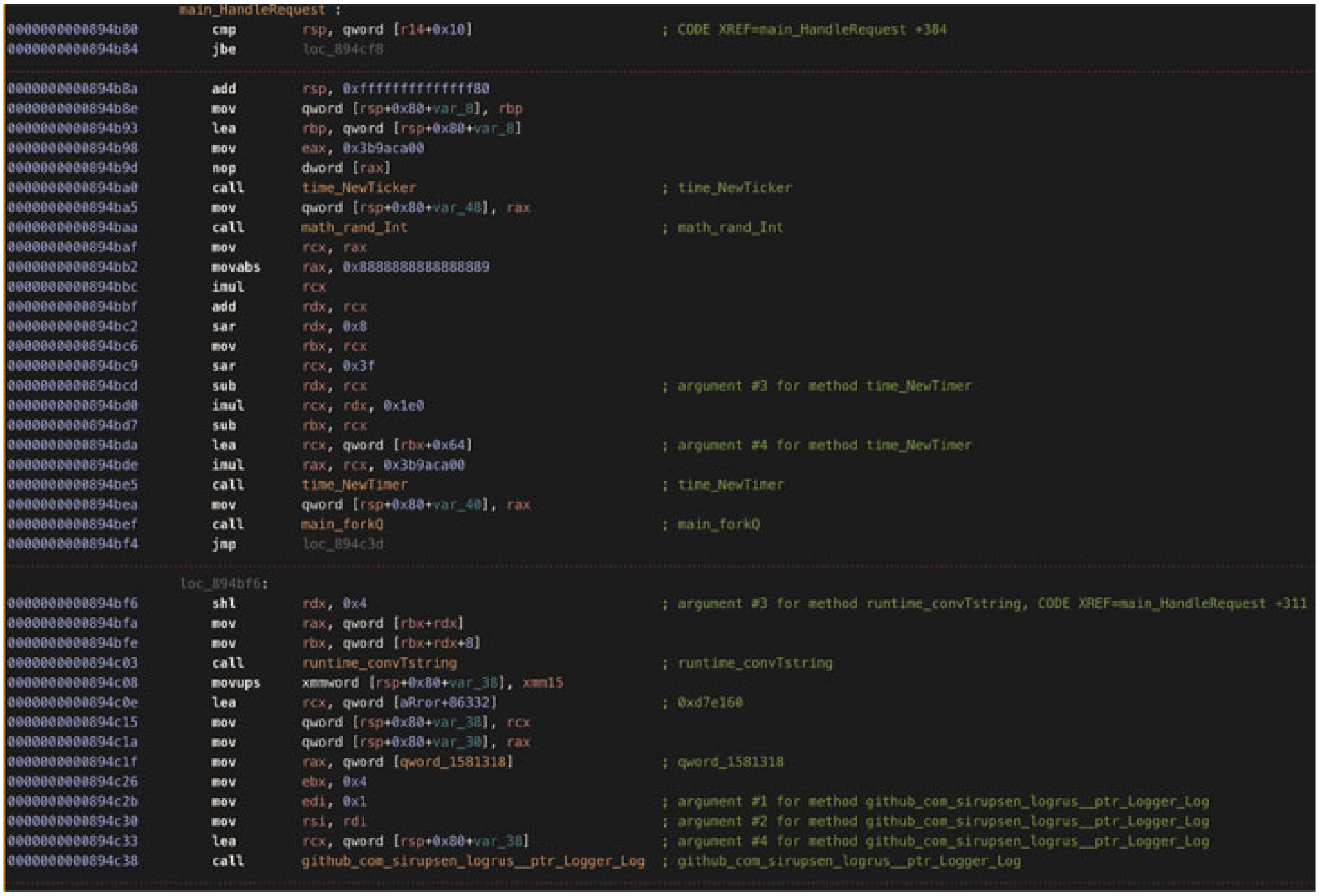
April 7, 2022

A first-of-its-kind malware targeting Amazon Web Services' (AWS) Lambda serverless computing platform has been discovered in the wild.

Dubbed "Denonia" after the name of the domain it communicates with, "the malware uses newer address resolution techniques for command and control traffic to evade typical detection measures and virtual network access controls," Cado Labs researcher Matt Muir said.

The artifact analyzed by the cybersecurity company was uploaded to the VirusTotal database on February 25, 2022, sporting the name "python" and packaged as a 64-bit ELF executable.

However, the filename is a misnomer, as Denonia is programmed in Go and harbors a customized variant of the XMRig cryptocurrency mining software. That said, the mode of initial access is unknown, although it's suspected it may have involved the compromise of AWS Access and Secret Keys.



```
                            main_HandleRequest :
0000000000894b80        cmp        rsp, qword [r14+0x10]                    ; CODE XREF=main_HandleRequest +384
0000000000894b84        jbe        loc_894cf8

0000000000894b8a        add        rsp, 0xffffffffffffff80
0000000000894b8e        mov        qword [rsp+0x80+var_8], rbp
0000000000894b93        lea        rbp, qword [rsp+0x80+var_8]
0000000000894b98        mov        eax, 0x3b9aca00
0000000000894b9d        nop        dword [rax]
0000000000894ba0        call       time_NewTicker                           ; time_NewTicker
0000000000894ba5        mov        qword [rsp+0x80+var_48], rax
0000000000894baa        call       math_rand_Int                            ; math_rand_Int
0000000000894baf        mov        rcx, rax
0000000000894bb2        movabs     rax, 0x8888888888888889
0000000000894bbc        imul       rcx
0000000000894bbf        add        rdx, rcx
0000000000894bc2        sar        rdx, 0x8
0000000000894bc6        mov        rbx, rcx
0000000000894bc9        sar        rcx, 0x3f
0000000000894bcd        sub        rdx, rcx                                 ; argument #3 for method time_NewTimer
0000000000894bd0        imul       rcx, rdx, 0x1e0
0000000000894bd7        sub        rbx, rcx
0000000000894bda        lea        rcx, qword [rbx+0x64]                    ; argument #4 for method time_NewTimer
0000000000894bde        imul       rax, rcx, 0x3b9aca00
0000000000894be5        call       time_NewTimer                            ; time_NewTimer
0000000000894bea        mov        qword [rsp+0x80+var_40], rax
0000000000894bef        call       main_forkQ                               ; main_forkQ
0000000000894bf4        jmp        loc_894c3d

                            loc_894bf6:
0000000000894bf6        shl        rdx, 0x4                                 ; argument #3 for method runtime_convTstring, CODE XREF=main_HandleRequest +311
0000000000894bfa        mov        rax, qword [rbx+rdx]
0000000000894bfe        mov        rbx, qword [rbx+rdx+8]
0000000000894c03        call       runtime_convTstring                      ; runtime_convTstring
0000000000894c08        movups     xmmword [rsp+0x80+var_38], xmm15
0000000000894c0e        lea        rcx, qword [aRror+86332]                 ; 0xd7e160
0000000000894c15        mov        qword [rsp+0x80+var_38], rcx
0000000000894c1a        mov        qword [rsp+0x80+var_30], rax
0000000000894c1f        mov        rax, qword [qword_1581318]               ; qword_1581318
0000000000894c26        mov        ebx, 0x4
0000000000894c2b        mov        edi, 0x1                                 ; argument #1 for method github_com_sirupsen_logrus__ptr_Logger_Log
0000000000894c30        mov        rsi, rdi                                 ; argument #2 for method github_com_sirupsen_logrus__ptr_Logger_Log
0000000000894c33        lea        rcx, qword [rsp+0x80+var_38]             ; argument #4 for method github_com_sirupsen_logrus__ptr_Logger_Log
0000000000894c38        call       github_com_sirupsen_logrus__ptr_Logger_Log ; github_com_sirupsen_logrus__ptr_Logger_Log
```

Another notable feature of the malware is its use of DNS over HTTPS (DoH) for communicating with its command-and-control server ("gw.denonia[.]xyz") by concealing the traffic within encrypted DNS queries.

In a statement shared with The Hacker News, Amazon stressed that "Lambda is secure by default, and AWS continues to operate as designed," and that users violating its acceptable use policy (AUP) will be prohibited from using its services.

While Denonia has been clearly designed to target AWS Lambda since it checks for Lambda environment variables prior to its execution, Cado Labs also found that it can be run outside of it in a standard Linux server environment.

"The software described by the researcher does not exploit any weakness in Lambda or any other AWS service," the company said. "Since the software relies entirely on fraudulently obtained account credentials, it is a distortion of facts to even refer to it as malware because it lacks the ability to gain unauthorized access to any system by itself."

However, "python" isn't the only sample of Denonia unearthed so far, what with Cado Labs finding a second sample (named "bc50541af8fe6239f0faa7c57a44d119.virus") that was uploaded to VirusTotal on January 3, 2022.

"Although this first sample is fairly innocuous in that it only runs crypto-mining software, it demonstrates how attackers are using advanced cloud-specific knowledge to exploit complex cloud infrastructure, and is indicative of potential future, more nefarious attacks," Muir said.

SHARE □ □ □ □ ךָ
SHARE □