# A crowning achievement: Exploring the exploit of Royal ransomware

logpoint.com/en/blog/exploring-the-exploit-of-royal-ransomware/

Rasmus Plambech January 5, 2023

***By Anish Bogati, Security Research***

**Contents**



## TL;DR

First observed in January 2022 and unlike any other ransomware we have covered, Royal is a private group with no known affiliations at this time. In another campaign, initial access is gained via “callback” phishing attacks. In this type of attack, the threat actors send an email containing a message to update a subscription of some kind and make the victim call the given number. When victims make a call mentioned in the mail, threat actors social engineered them to download and install their malware. A similar technique was previously used by the Conti group. The gang has been seen employing virtual hard disk (VHD) files that seem like legitimate software to speed up the transfer of first-stage payloads.

** *Get research and analysis, insight, plus hints and tips, on how to detect, manage, and respond to Royal ransomware in the main blog.*

*Head to the contents and click each section for quick navigation.*

*First observed in January 2022 and unlike any other ransomware we have covered, Royal is a private group with no known affiliations at this time. However, Royal has been found deployed by threat actors DEV-0569. The group has been found using Google ads to redirect users to forums, posts, and blog comments, or sending phishing emails that contain links to download the malware.*


Royal leak site

In another campaign, initial access is gained via “callback” phishing attacks. In this type of attack, the threat actors send an email containing a message to update a subscription of some kind and make the victim call the given number. When victims make a call mentioned in the mail, threat actors social engineered them to download and install their malware. A similar technique was previously used by the Conti group.

To bypass defense, DEV-0569 employs phishing links and malicious downloaders that seem like legitimate installers or updates. The gang has been seen employing virtual hard disk (VHD) files that seem like legitimate software to speed up the transfer of first-stage payloads.

## Fast Facts

- Since November 2022, the data of more than 60 victims has been leaked
- Written in C++ and is specifically designed for the Windows OS
- Supports various file encryption options
- Uses various social engineering techniques for initial access
- Uses the RestartManager to shut down the process that is using the files to be encrypted

According to the U.S. Department of HHS, after gaining initial access and execution of their payload, the Cobalt Strike beacon is dropped in the victim system to maintain persistence, access credentials, and move laterally. The main ransomware payload uses a few command line arguments like “-id” and “-path.” Without using those command line arguments and corresponding values, it only performs system information discovery and inhibits system recovery by deleting all the shadow

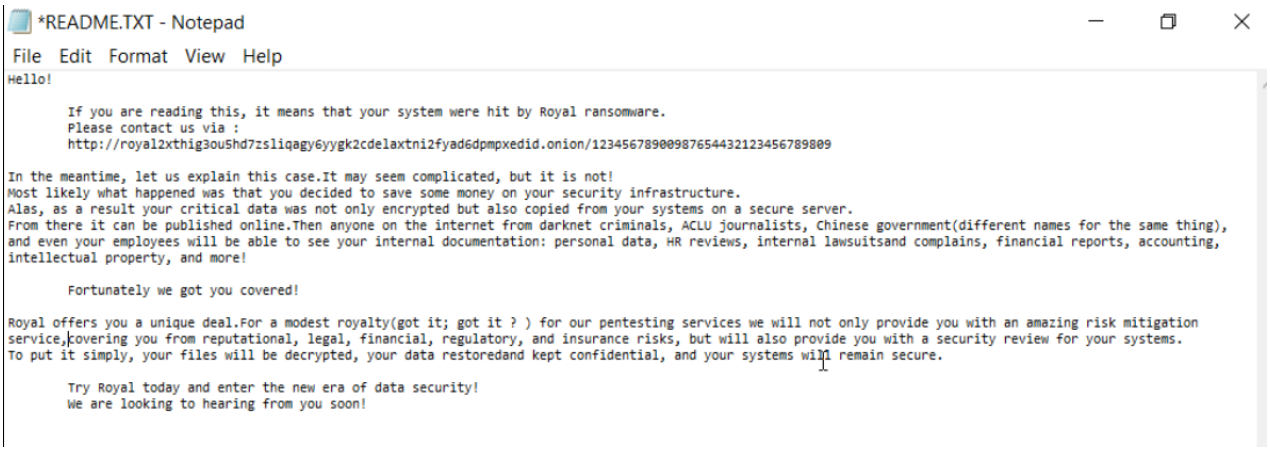
copies as a method of evading sandboxing and analysis. The third argument is ”-ep” and is not mandatory. The encrypted files are appended with the “.royal” file extension. While encrypting files it avoids encrypting specific folders and file types to prevent a system crash.

Below are the file types that are not encrypted by Royal.

> **.exe, .dll, .bat, .lnk, .royal**

Below are those folders whose sub-folders and files are not encrypted by the ransomware.

> **Windows, Royal, Perflogs, Tor browser, Boot, $recycle.bin, Windows.old, $window.~ws, $windows.~bt, Mozilla, Google**




Ransom note

## Royal analysis

We have used multiple ransomware variants for our analysis to provide an all-encompassing detection and understanding. The samples we used were retrieved from MalwareBazaar and for reference, we also used samples from tria.ge sandboxes. We performed a static and dynamic analysis in our sandbox which is running Microsoft Windows 10 Enterprise. The outcomes of our analysis are explained below.

To execute the malware, currently, two things are mandatory: “-path” and “-id.” The path argument is for encrypting the specified directory and the “-id” argument is required for starting the encryption process and it should be a unique 32-character string. We have also observed the third argument “-ep” which seems to be optional for now and is used to determine the percentage of files that is to be encrypted. After analyzing the sample we have found out that if the command line arguments are missing or the parameter’s value is not correct, then it doesn’t perform encryption of files.

```
if (0 < local_6e98[0]) {
  do {
    iVar2 = lstrcmpW(*ppWVar5,L"-path");
    iVar6 = (int)pWVar7;
    if (iVar2 == 0) {
      pWVar8 = ppWVar5[1];
      iVar6 = iVar6 + 1;
      ppWVar5 = ppWVar5 + 1;
    }
    else {
      iVar2 = lstrcmpW(*ppWVar5,L"-id");
      if (iVar2 == 0) {
        pWVar7 = ppWVar5[1];
        ppWVar5 = ppWVar5 + 1;
        iVar6 = iVar6 + 1;
        iVar2 = lstrlenW(pWVar7);
        WideCharToMultiByte(0xfde9,0,pWVar7,iVar2,local_268,0x21,(LPCSTR)0x0,(LPBOOL)0x0);
      }
      else {
        iVar2 = lstrcmpW(*ppWVar5,L"-ep");
        if (iVar2 == 0) {
          ppWVar1 = ppWVar5 + 1;
          ppWVar5 = ppWVar5 + 1;
          iVar6 = iVar6 + 1;
          lVar3 = _wtol(*ppWVar1);
          if (99 < lVar3 - 1U) {
            lVar3 = 0x32;
```

Showing command-line arguments

The malware retrieves the command-line string for the process using the **GetCommandLineW** API and **CommandLineToArgvW** is used to obtain an array of pointers to the command-line arguments.

```
call    cs:GetCommandLineW
mov     rcx, rax
lea     rdx, [rsp+6EF0h+var_6E88]
call    cs:CommandLineToArgvW
```

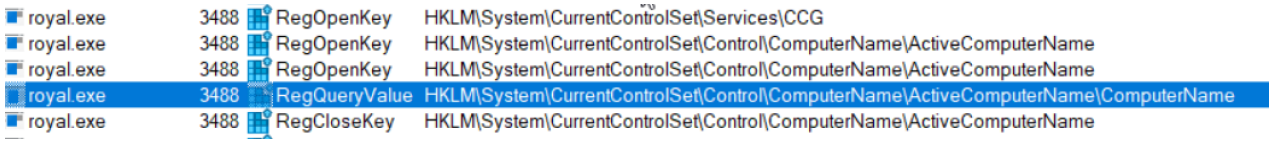
Function used to retrieve command line

We have also found a call to **GetNativeSystemInfo** API which contains various system information like architecture, number of processors, and page size. Using the **GetNativeSystemInfo** with **dwNumberOfProcessor** function, the malware is trying to retrieve the number of the processor's cores.

```
lea     rcx, [rsp+78h+SystemInfo] ; lpSystemInfo
call    cs:GetNativeSystemInfo
mov     eax, [rsp+78h+SystemInfo.dwNumberOfProcessors]
```

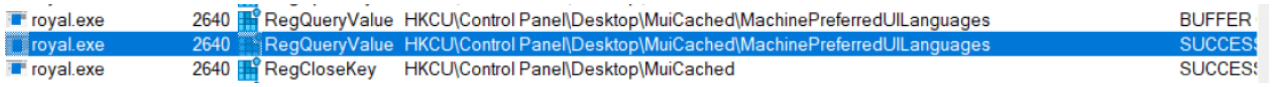
Function used to retrieve systeminfo

We have also found the instance where Royal was querying the registry key “HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ComputerName\ActiveComputerName” to retrieve the system name.




Querying system name

Also Royal has been found querying the registry key shown in below image to check system language settings. This can be done to verify language settings and display output accordingly.




Querying system language

Royal uses the GetIpAddrTable function to retrieve the IP address table from the local computer. It returns a list of IP addresses and their corresponding subnet masks and default gateways.

```
lea     rdx, [rsp+78h+pdwSize] ; pdwSize
xor     ecx, ecx              ; pIpAddrTable
call    cs:GetIpAddrTable
```

Function used to retrieve IP address information

The malware also uses the Windows API function NetShareEnum to enumerate the shared resources on the network. The below function compares the names of the shared resources returned by the API call with the strings “ADMIN$” and “IPC$” to prevent the malware from encrypting those shares.

```
WSAAddressToStringW(&local_288,0x10,0,local_278);
do {
  local_298 = 0;
  local_28c = 0;
  local_290 = 0;
  local_2a0 = (LPCWSTR *)0x0;
  DVar1 = NetShareEnum(local_278,1,(char *)&local_2a0,0xffff,(ushort *)&local_298,
                       (ushort *)&local_28c);
  if ((DVar1 != 0) && (DVar1 != 0xea)) break;
  uVar4 = 1;
  ppWVar5 = local_2a0;
  if (local_298 != 0) {
    do {
      iVar2 = lstrcmpiW(L"ADMIN$",*ppWVar5);
      if ((iVar2 != 0) && (iVar2 = lstrcmpiW(L"IPC$",*ppWVar5), iVar2 != 0)) {
        wsprintfW(local_248,L"\\\\%s\\%s");
        local_2c8[0] = 0;
        uVar3 = 0xffffffffffffffff;
        local_2b8 = 0;
        local_2b0 = 7;
        do {
          uVar3 = uVar3 + 1;
        } while (local_248[uVar3] != L'\0');
        if (uVar3 < 8) {
          local_2b8 = uVar3;
          FUN_1401e3fa0(local_2c8,local_248);
          *(undefined2 *)((longlong)local_2c8 + uVar3 * 2) = 0;
        }
        else {
          FUN_14007cd00(local_2c8);
        }
        FUN_14007c000(*(undefined8 *)(param_1 + 0x6020));
      }
      uVar4 = uVar4 + 1;
      ppWVar5 = ppWVar5 + 3;
    } while (uVar4 <= local_298);
  }
  NetApiBufferFree(local_2a0);
} while (DVar1 == 0xea);
```

Function used to query network share

To prevent system recovery, Royal has been using the vssadmin utility to delete all the available shadow copies without displaying any message.

```
007DE64:                                ; CODE XREF: WinMain+97↑j
vind { // sub_1401E2A60
        xor     edx, edx
        lea     rcx, [rbp+6DF0h+var_230]
        mov     r8d, 200h
        call    sub_1401E4650
        lea     rdx, aDeleteShadowsA ; " delete shadows /all /quiet"
        lea     rcx, [rbp+6DF0h+var_230]
        call    cs:wsprintfW
        xorps   xmm0, xmm0
        mov     [rsp+6EF0h+var_6E80], 68h ; 'h'
        xor     eax, eax
        lea     rdx, [rbp+6DF0h+var_230]
        mov     [rbp+6DF0h+var_6E1C], eax
        lea     rcx, aCWindowsSystem ; "C:\\Windows\\System32\\vssadmin.exe"
        mov     [rsp+6EF0h+var_6E90], rax
```

Displaying process creation of vssadmin

While executing the malware we observed the malware spawning child process for running vssadmin to delete all shadow copies.

- C:\Users\Admin\AppData\Local\Temp\f484f919ba6e36ff33e4fb391b8859a94d89c172a465964f99d6113b55ced429.exe

```
C:\Users\Admin\AppData\Local\Temp\f484f919ba6e36ff33e4fb391b8859a94d89c172a465964f99d6113b55ced429.ex
e -path C:\ -id 1234456789098765432123456789O987
```

- C:\Windows\System32\vssadmin.exe

```
delete shadows /all /quiet
```

Shadow copy delete via vssadmin

Before starting up the encryption process, the malware checks if any of the targeted files to be encrypted are being used by other processes. If the files are found to be used by other applications, the malware uses RestartManager to kill those applications and services using the resource.

| | | | | |
|---|---|---|---|---|
| royal.exe | 4880 | RegCreateKey | HKCU\Software\Microsoft\RestartManager\Session0001 | SUCCESS |
| royal.exe | 4880 | RegQueryKey | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001 | SUCCESS |
| royal.exe | 4880 | RegEnumValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001 | SUCCESS |
| royal.exe | 4880 | RegDeleteValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001\RegFilesHash | SUCCESS |
| royal.exe | 4880 | RegEnumValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001 | SUCCESS |
| royal.exe | 4880 | RegDeleteValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001\RegFiles0000 | SUCCESS |
| royal.exe | 4880 | RegEnumValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001 | SUCCESS |
| royal.exe | 4880 | RegDeleteValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001\Sequence | SUCCESS |
| royal.exe | 4880 | RegEnumValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001 | SUCCESS |
| royal.exe | 4880 | RegDeleteValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001\SessionHash | SUCCESS |
| royal.exe | 4880 | RegEnumValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001 | SUCCESS |
| royal.exe | 4880 | RegDeleteValue | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001\Owner | SUCCESS |
| royal.exe | 4880 | RegCloseKey | HKCU\SOFTWARE\Microsoft\RestartManager\Session0001 | SUCCESS |

Application session delete

After freeing up the resources, the malware begins the encryption process. While viewing the events through ProcMon, we observed the following actions while a file was being encrypted.

| | | | |
|---|---|---|---|
| royal.exe | CreateFile | C:\Users\tutaans\Desktop\floss opt.txt | Desired Access: All Access, Disposition: Open, Options: Synchronous IO Non-Al.. |
| royal.exe | QueryStandardInformationFile | C:\Users\tutaans\Desktop\floss opt.txt | AllocationSize: 393,216, EndOfFile: 391,460, NumberOfLinks: 1, DeletePending: F.. |
| royal.exe | WriteFile | C:\Users\tutaans\Desktop\floss opt.txt | Offset: 391,460, Length: 540, Priority: Normal |
| royal.exe | SetEndOfFileInformationFile | C:\Users\tutaans\Desktop\floss opt.txt | EndOfFile: 392,000 |
| royal.exe | SetAllocationInformationFile | C:\Users\tutaans\Desktop\floss opt.txt | AllocationSize: 392,000 |
| royal.exe | ReadFile | C:\Users\tutaans\Desktop\floss opt.txt | Offset: 0, Length: 391,472 |
| royal.exe | WriteFile | C:\Users\tutaans\Desktop\floss opt.txt | Offset: 0, Length: 391,472, Priority: Normal |
| royal.exe | WriteFile | C:\Users\tutaans\Desktop\floss opt.txt | Offset: 391,472, Length: 512 |
| royal.exe | WriteFile | C:\Users\tutaans\Desktop\floss opt.txt | Offset: 391,984, Length: 8 |
| royal.exe | WriteFile | C:\Users\tutaans\Desktop\floss opt.txt | Offset: 391,992, Length: 8 |
| royal.exe | FlushBuffersFile | C:\Users\tutaans\Desktop\floss opt.txt | |
| royal.exe | WriteFile | C:\Users\tutaans\Desktop\floss opt.txt | Offset: 0, Length: 393,216, I/O Flags: Non-cached, Paging I/O, Synchronous Pagin.. |
| royal.exe | CloseFile | C:\Users\tutaans\Desktop\floss opt.txt | |
| royal.exe | CreateFile | C:\Users\tutaans\Desktop\floss opt.txt | Desired Access: Read Attributes, Delete, Synchronize, Disposition: Open, Option.. |
| royal.exe | QueryAttributeTagFile | C:\Users\tutaans\Desktop\floss opt.txt | Attributes: A, ReparseTag: 0x0 |
| royal.exe | QueryBasicInformationFile | C:\Users\tutaans\Desktop\floss opt.txt | CreationTime: 11/29/2022 9:39:06 AM, LastAccessTime: 11/29/2022 9:48:45 AM, L.. |
| royal.exe | SetRenameInformationFile | C:\Users\tutaans\Desktop\floss opt.txt | ReplaceIfExists: False, FileName: C:\Users\tutaans\Desktop\floss opt.txt.royal |
| royal.exe | CloseFile | C:\Users\tutaans\Desktop\floss opt.txt.ro... | |

Encryption process view form ProcMon without “-ep” argument

## Analysis of an older version of Royal

As mentioned, the Royal operator was previously using malware known as Zeon. We were also able to observe the sample analysis in tria.ge. After executing this malware sample it first tries to stop various services like backup, web engine, POP3, IMAP, SQL, VSS, antivirus, etc., using net and net1 binary. It also attempts to kill various processes relating to Office products, Firefox, Wordpad, calculator, oracle, and steam using taskkill.exe binary.

- C:\Windows\SysWOW64\net.exe
  ```
  net stop /y backup
  ```
  - C:\Windows\SysWOW64\net1.exe
    ```
    C:\Windows\system32\net1 stop /y backup
    ```
- C:\Windows\SysWOW64\net.exe
  ```
  net stop /y wbengine
  ```
  - C:\Windows\SysWOW64\net1.exe
    ```
    C:\Windows\system32\net1 stop /y wbengine
    ```
- C:\Windows\SysWOW64\net.exe
  ```
  net stop /y McShield
  ```
  - C:\Windows\SysWOW64\net1.exe
    ```
    C:\Windows\system32\net1 stop /y McShield
    ```
- C:\Windows\SysWOW64\taskkill.exe
  ```
  taskkill /im steam.exe /f
  ```
- C:\Windows\SysWOW64\net.exe
  ```
  net stop /y mfefire
  ```
  - C:\Windows\SysWOW64\net1.exe
    ```
    C:\Windows\system32\net1 stop /y mfefire
    ```
- C:\Windows\SysWOW64\net.exe
  ```
  net stop /y EhttpSrv
  ```
  - C:\Windows\SysWOW64\net1.exe
    ```
    C:\Windows\system32\net1 stop /y EhttpSrv
    ```
- C:\Windows\SysWOW64\net.exe
  ```
  net stop /y KAVF
  ```
  - C:\Windows\SysWOW64\net1.exe
    ```
    C:\Windows\system32\net1 stop /y KAVF
    ```
- C:\Windows\SysWOW64\taskkill.exe
  ```
  taskkill /im ocautoupds.exe /f
  ```
- C:\Windows\SysWOW64\net.exe
  ```
  net stop /y VeeamNFSSvc
  ```
  - C:\Windows\SysWOW64\net1.exe
    ```
    C:\Windows\system32\net1 stop /y VeeamNFSSvc
    ```
- C:\Windows\SysWOW64\taskkill.exe
  ```
  taskkill /im backup.exe /f
  ```

Royal process tree

Then the malware attempts to create a scheduled task using schtask.exe with a random task name. In this case, Royal uses a binary masquerading as an image to evade detection. To overcome any failed attempt to schedule a task, the malware tries to schedule the task multiple times.

C:\Windows\SysWOW64\cmd.exe

```
C:\Windows\system32\cmd.exe /c schtasks.exe /Create /TN zE0xO6bGus /TR "CMD.EXE DEL /F /Q "C:\Pr
ogramData\pqBxGx.jpg" >> NUL" /sc once /st 00:00 /RL HIGHEST
```

C:\Windows\SysWOW64\schtasks.exe

```
schtasks.exe  /Create /TN zE0xO6bGus /TR "CMD.EXE DEL /F /Q "C:\ProgramData\pqBxGx.jpg" >> NU
L" /sc once /st 00:00 /RL HIGHEST
```

Scheduled task creation

After scheduling the task, the malware uses the schtasks.exe binary to execute the task.

C:\Windows\SysWOW64\cmd.exe

```
C:\Windows\system32\cmd.exe /c schtasks.exe /Run /TN zE0xO6bGus
```

C:\Windows\SysWOW64\schtasks.exe

```
schtasks.exe  /Run /TN zE0xO6bGus
```

Scheduled task execution

Then the below command is used to delete the file “pqBxGx.jpg.” The “/F” option forces the deletion of the file, even if it is read-only. The “/Q” option tells the command to delete the file without asking for confirmation. The “>> NUL” portion redirects the output of the command to the null device, which discards it and prevents it from being displayed on the screen.

C:\Windows\system32\CMD.EXE

```
CMD.EXE DEL /F /Q C:\ProgramData\pqBxGx.jpg >> NUL
```

Command to delete pqBxGx.jpg file.

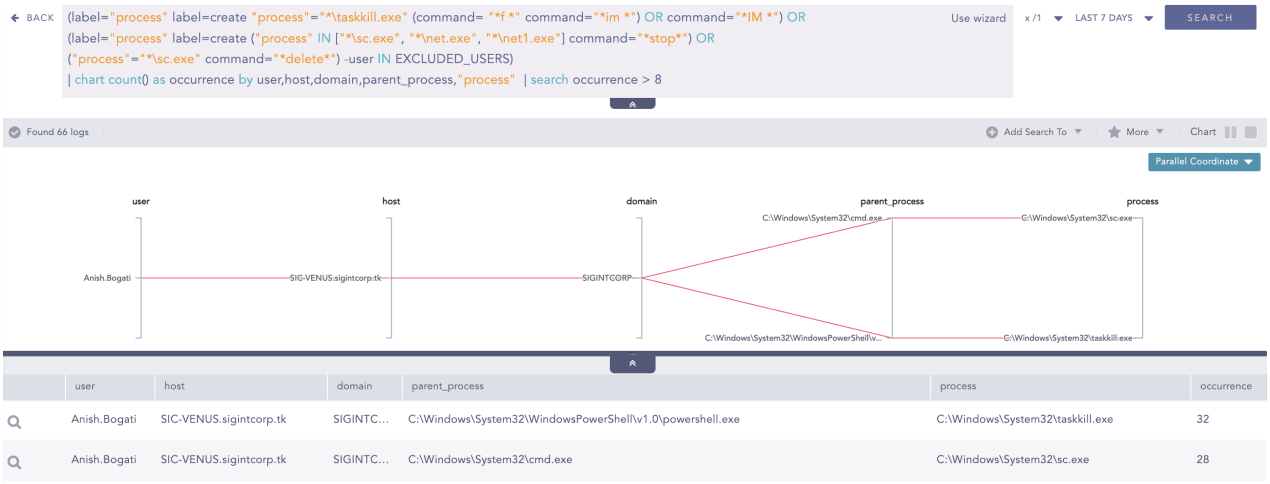
## Detecting Royal using Logpoint

### Log Source Needed

- Windows Event Logs
- Sysmon for Windows

As mentioned earlier, to set up the precondition for the ransomware to detonate, the malware stops many services and kills various processes. To detect such activity of Royal ransomware, analysts can use the below query.

```
(label="process" label=create "process"="*\taskkill.exe"
(command= "*f *" command="*im *") OR command="*IM *") OR
(label="process" label=create ("process" IN ["*\sc.exe", "*\net.exe", "*\net1.exe"]
command="*stop*") OR ("process"="*\sc.exe" command="*delete*") -user IN EXCLUDED_USERS)
| chart count() as occurrence by user, host, domain,"process",parent_process
| search occurrence > 8
```




Detecting Royal activity in Logpoint

Adversaries have used scheduled task functionality to facilitate for single or repetitive execution of malicious codes. Here, Royal ransomware has been seen to create a scheduled task to launch the ransomware. Thus, analysts need to look for the creation of scheduled tasks using the schtasks binary.

```
(label="Process" label=Create "process"="*\schtasks.exe" command="* /create *")
OR (label="Registry" label="Key" label="Map" event_type=CreateKey
"target_object"="*\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tree\*"
 -target_object IN
["*\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\Schedule\TaskCache\Tree\Microsoft\Windows\UpdateOrchestrator*"])
```

So, not just the creation of scheduled tasks, but events related to running of the scheduled task through schtasks should be monitored too.

```
label="Process" label=Create "process"="*\schtasks.exe" command="* /run *"
```

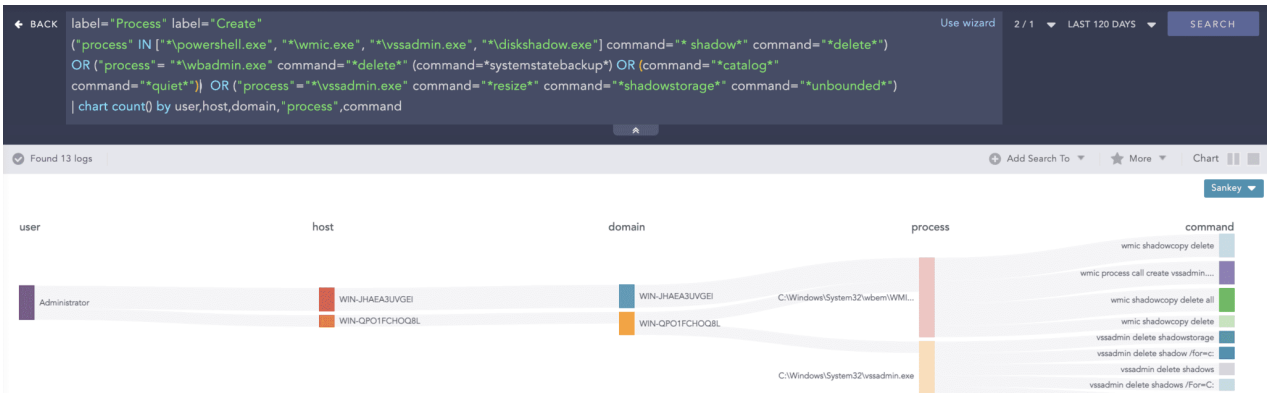
The malware also performs SMB share enumeration to encrypt the share folder, so access to multiple share folders in a short span from the same user and host needs to be monitored.

```
[5 norm_id=winserver event_id=5140 share_name=*
having same user,host,source_address within 1 minute]
```

According to the requirement, the number of events and time duration can be changed in the above query.

To prevent recovery of the drives, Royal has been found deleting the volume shadow copy of the drive using the vssadmin utility.

```
label="Process" label="Create"
("process" IN ["*\powershell.exe", "*\wmic.exe", "*\vssadmin.exe", "*\diskshadow.exe"]
command="* shadow*" command="*delete*") OR
("process"= "*\wbadmin.exe" command="*delete*" (command=*systemstatebackup*)
OR (command="*catalog*" command="*quiet*"))
OR ("process"="*\vssadmin.exe" command="*resize*" command="*shadowstorage*"
command="*unbounded*")
```




Detecting Royal activity in Logpoint

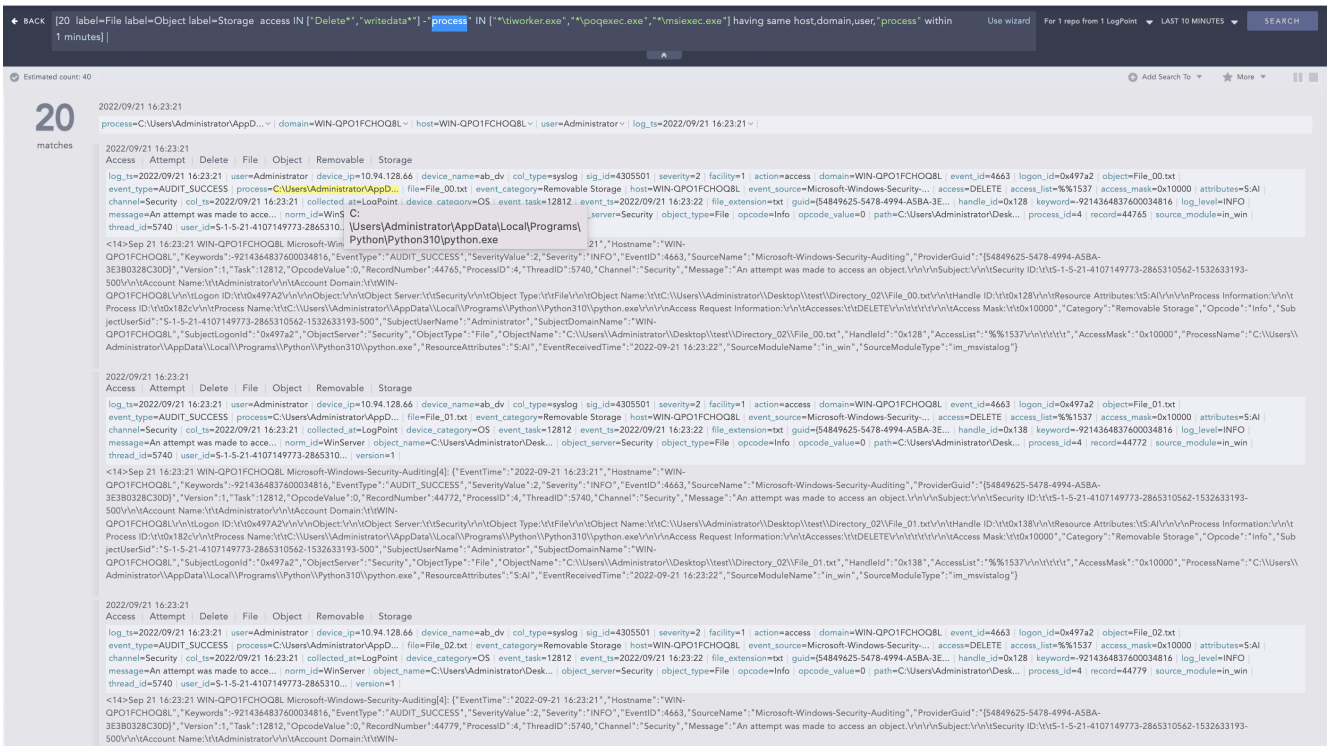
If files to be encrypted are being used by other applications, Royal uses RestartManager to kill the various processes that are using the files to be encrypted. The below query detects five such events that occurred within a minute.

```
[5 norm_id=Windowssysmon event_id=12 event_type=DeleteValue
target_object="*\software\microsoft\restartmanager\session*" within 1 minute]
```

To detect the relevant logs, registry auditing for the RestartManager registry key needs to be enabled in sysmon.

High Volume of File Modification or Deletion in Short Span:

```
[30 label=File label=Object label=Storage access IN ["Delete*","writedata*"]
-"process" IN ["*\tiworker.exe","*\poqexec.exe","*\msiexec.exe"] having same host,
domain, user," process" within 1 minute]
```



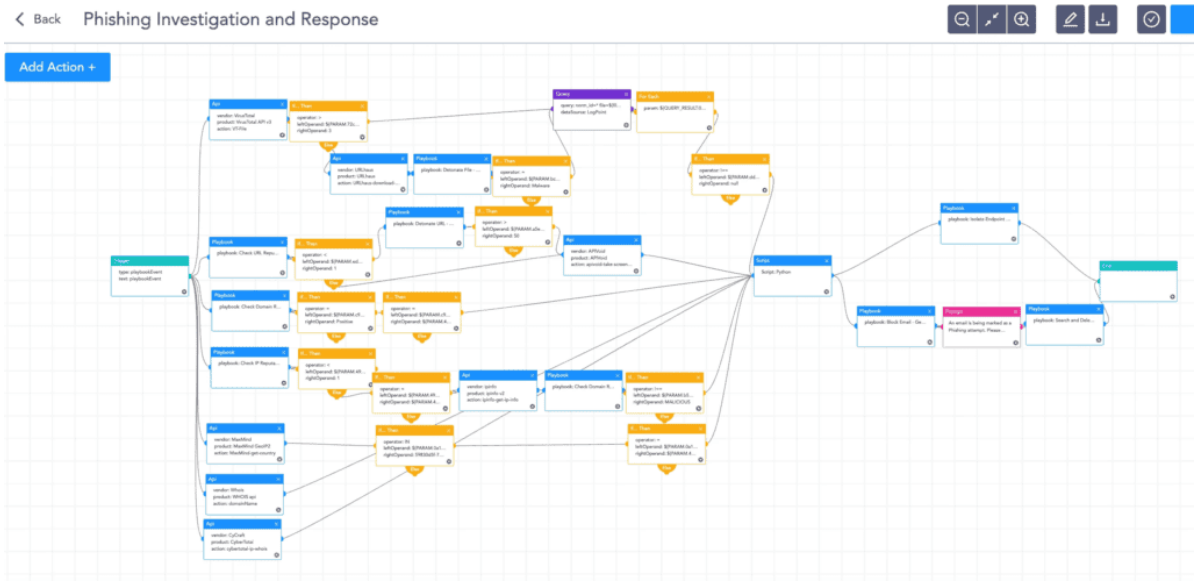
The above image shows that the Python process has modified or deleted 20 files in a minute. Depending on the situation and the needs, the number of logs and the time range to trigger alerts can be modified. This alert detects a large number of file modifications or deletions in a short period so, it can detect file encryption activity by the ransomware.

The given alerts are available in Logpoint's latest Alert Rules package and are available free for download for Logpoint customers.

## Investigation and response using Logpoint

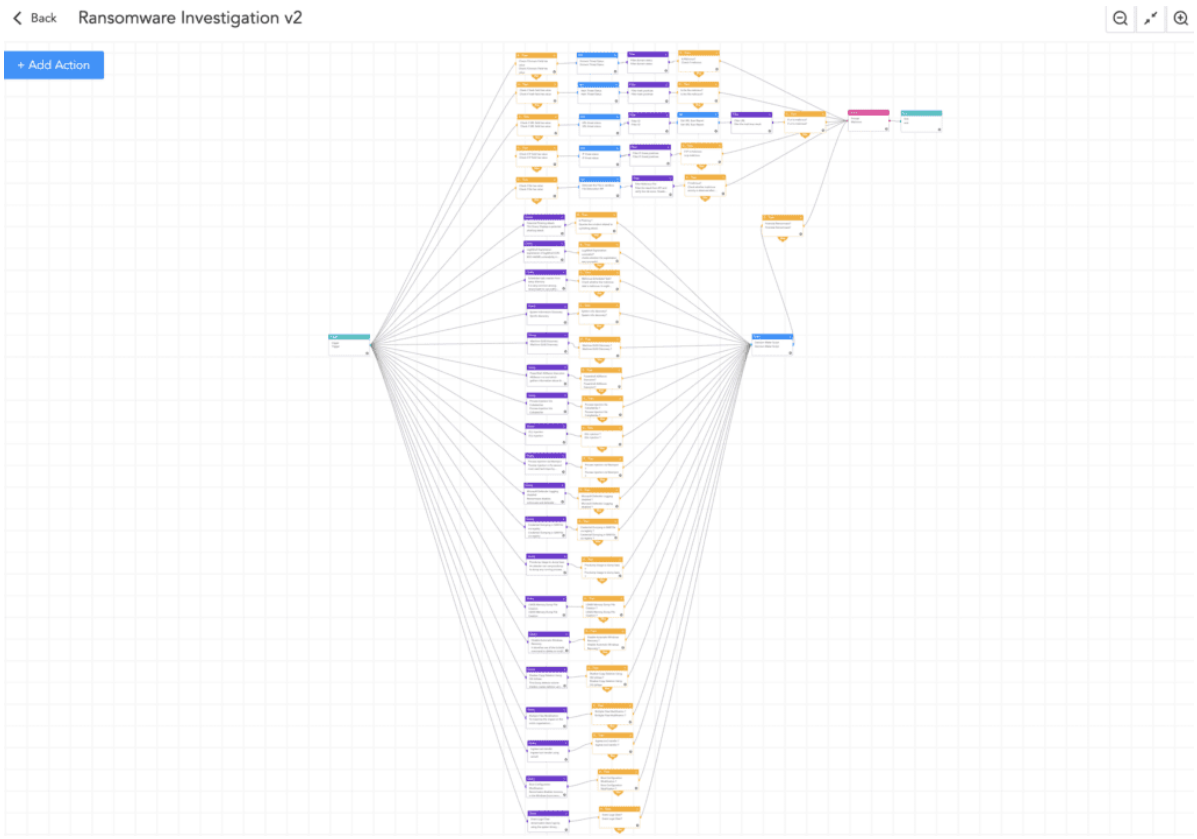
There is no actual silver bullet for investigation and response to ransomware. However, Logpoint SOAR is by far the most useful tool for organizations to investigate and respond to ransomware attacks.

1. **Phishing investigation and response**
   Royal ransomware groups also use phishing techniques to gain initial access and this playbook ensures all suspicious phishing incidents are adequately investigated and responded to, dramatically reducing the response time and human error.



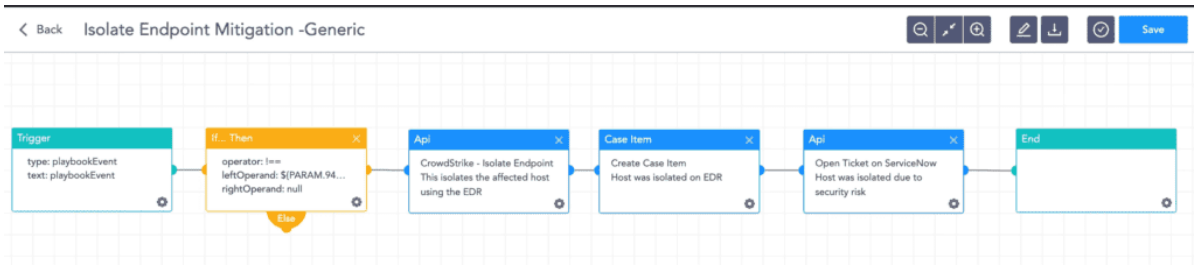

Phishing investigation playbook

2. **Ransomware investigation**
   This playbook thoroughly examines the IoCs and uses a sandbox to detonate the suspicious files. It also looks for the common TTPs used by the ransomware, improving the chances of detecting ransomware before it is too late. The playbook will prompt an alert message to the administrators if ransomware is identified, and will start further work to isolate the host and contain the malware.



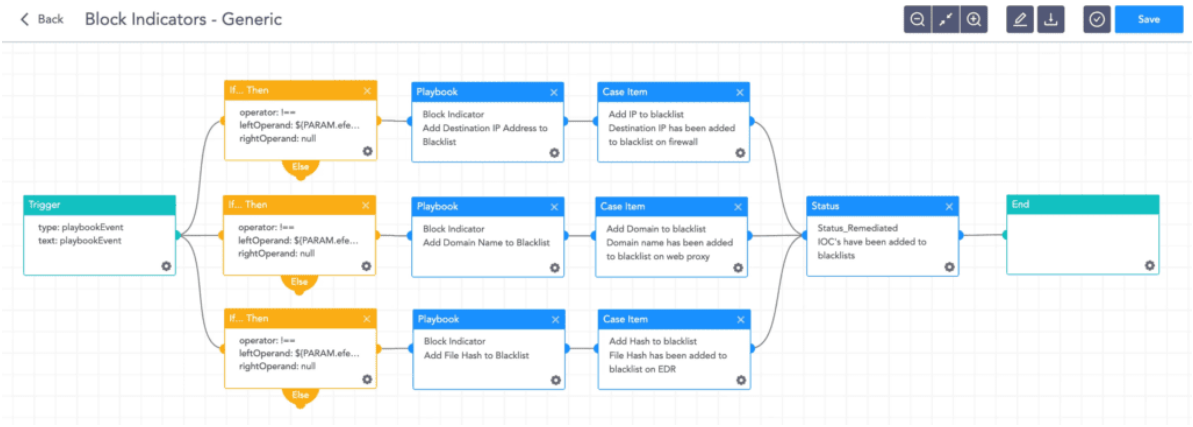

Ransomware investigation playbook

3. **Isolate endpoint mitigation**
   Using the new Logpoint AgentX, this playbook identifies the infected host and isolates it; and contains and quarantines it before it spreads to other machines. AgentX is a built-in response capability on endpoints and to start using it, contact your Logpoint representative.




Isolate endpoint mitigation playbook

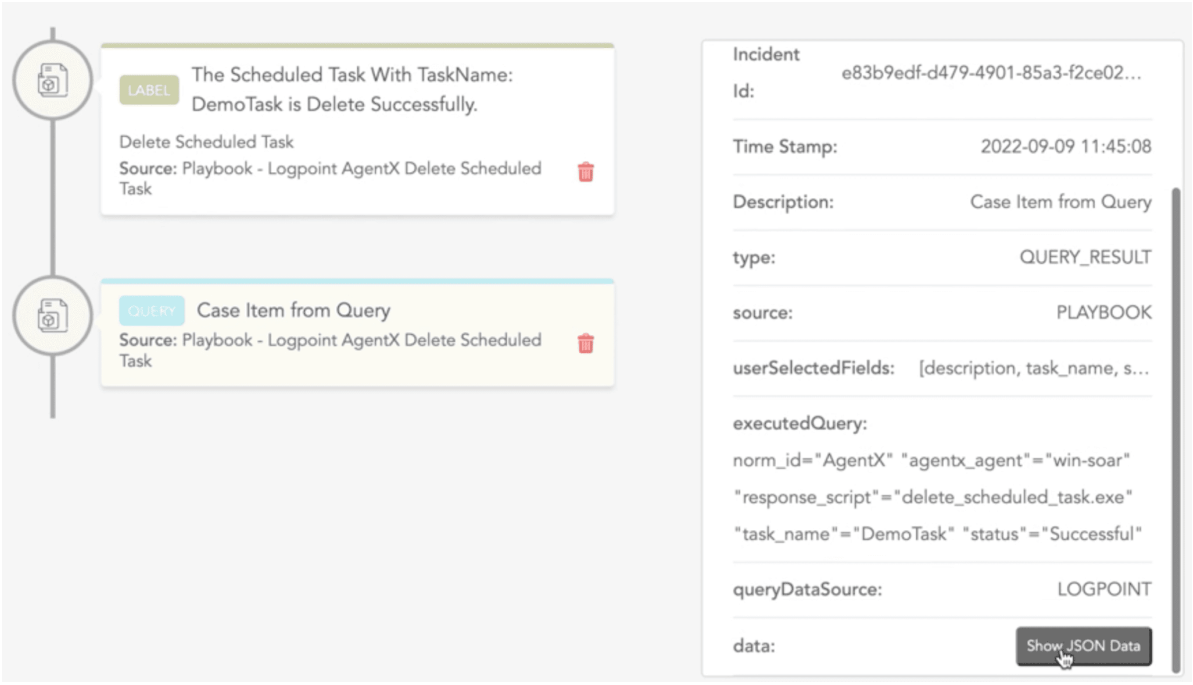
4. **Block indicators**
   This playbook checks if any IP, domain, URL, or host exists in a list of IoCs, blocks them, and adds them to the blocked list.



Block indicator playbook

5. **Delete scheduled task**

   The Logpoint AgentX delete scheduled task response playbook reduces the burden of manually deleting a suspicious scheduled task. This playbook requires the analyst to provide the hostname of the machine where the scheduled task was created, the manager IP address, and the scheduled task name.



Successful deletion of scheduled task

## Endpoint detection and remediation with AgentX

Logpoint AgentX is a lightweight application that transports logs and telemetry from endpoints (all servers, workstations, and applications) to the SIEM, and performs automated real-time investigation and remediation of threats with SOAR. With AgentX, security analysts get precise detection of

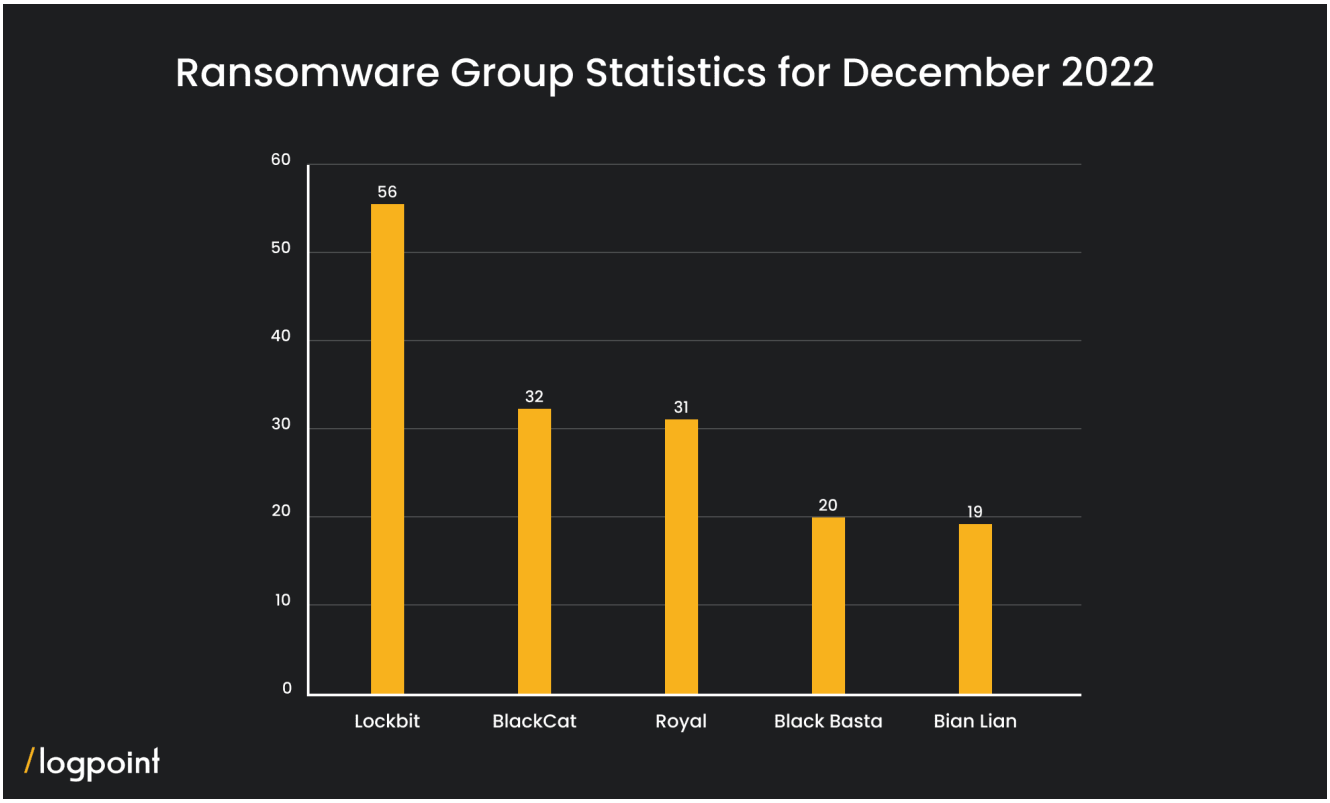
malicious malware and the ability to respond to threats in endpoints.

Logpoint AgentX is available now: Contact your representative.

## End-to-end detection, investigation, and response of Royal with Logpoint

Royal is a private group with no affiliates. Despite being a private group, Royal ransomware operators were able to leak the data of more than 60 victims on their leak site within a one-and-half-month period. This ransomware uses various tactics and techniques to achieve its end goals. For a faster encryption process, it uses partial encryption and other encryption options. As ransomware payloads are deployed in later stages, we can still detect them before they can encrypt an organization's sensitive data. Through proper system logging and using the Logpoint Converged SIEM platform combining SIEM, SOAR, and endpoint response, we can detect the traces of the ransomware and investigate and respond to the malware's activities.

Ransomware Group Statistics for December 2022




*Source: https://darkfeed.io/2023/01/02/ransomware-groups-statistics-december-2022/*

## Contact Logpoint

Contact us and learn why industry-leading companies choose Logpoint:

Contact Logpoint