Advanced Windows Firewall Subversion

Lin0xx of NoxusFiles

The Prevailing Mentality

- Applications aren't secure
- We have to cover for someone else's vulnerability
- Many people believe a firewall is the magic solution
  - Either HIPS or NIDS/IPS
- There's one small problem…

Theory

- If we exploit a vulnerable system, WE have execution control
- WE can do what the user can
- The attempts to stop us can be broken

Former Research

- Phrack 62 #13
- Presents some standard ways of bypassing many PFWs at the time and now
- Mostly involves abusing trust put in a certain process
- However, methods that use code injection without some extra techniques are caught

Tools Needed

- A debugger – ollydbg
- A disassembler – not totally necessary if we have a debugger that acts as one
  - IDA Pro
- An assembler – nasm
- A compiler – MSVC++ 2005 (free version works)

Windows Firewall

- Service Pack 2 has brought us a new and improved firewall
- Integrated into windows
- Offers a bit more control than the original
- One BIG problem…

Subverting the ICF

- Do a search on MSDN for “exercising the windows firewall”
  - Yeah.
- Has a COM interface exported
- All sorts of fun methods here:
  - put_FirewallEnabled – the one we want
- Can do other things such as arbitrarily authorize applications…

The Shellcode - FwExit

- fwMgr->get_LocalPolicy(&fwPolicy);
- fwPolicy->get_CurrentProfile(&fwProfile);
- fwProfile->put_FirewallEnabled(VARIANT_FALSE);

Demo

ZoneAlarm Personal Firewall

- One of the main HIPS used
- Has to defend from threats locally and remotely
- TrueVector
  - Driver
  - What attempts to keep ZoneAlarm safe from local threats
- Is quite aggressive about defending itself

Reversing ZoneAlarm

- Plan of attack:
  - Get write access to zlclient.exe process
  - Inject code that will disable it
  - Run the code
  - Make sure its done stealthily
- Call it ZaRooter

Finding the Process

- First, we need to find zlclient.exe's PID
- EnumProcesses() – Process list
- OpenProcess() with parameter of PROCESS_QUERY_INFORMATION
- EnumProcessModules() – gets the module handle of the process
- GetModuleBaseNameA() – gets the process name
  - We can then just strcmp() this to zlclient.exe

Big Problem

- TrueVector is a pest
- Behaves as a rootkit
- Hooks many system calls
  - This is why you can't just TerminateProcess() the firewall
- Most important
  - Hooks NtOpenProcess
- We can't get a handle to zlclient.exe

Enter the ZaRooter Driver

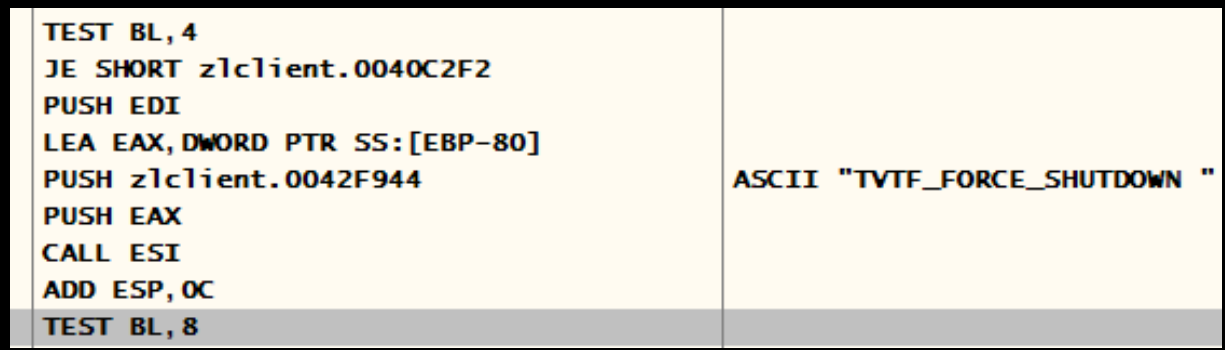
- TrueVector doesn't mind if we load a driver
- We can write to the SSDT
- This means we execute code on its level
  - We can un-hook every syscall that it has hooked

Pseudo-Rootkit Driver Code

- Create a MDL for the SSDT, which is..
- __declspec(dllimport) SSDT_Entry KeServiceDescriptorTable;
- Unhook NtOpenProcess and replace it with the address of the original
  - The original must be known ahead of time and differs between service packs
- hook_syscall(syscall_number, address);

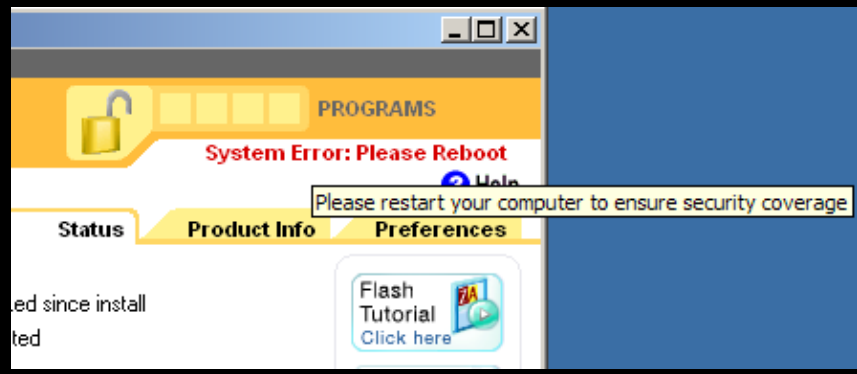
Shutdown Code

- Location: 0x0040c261
- Pass it a parameter of 4 to shut TrueVector down



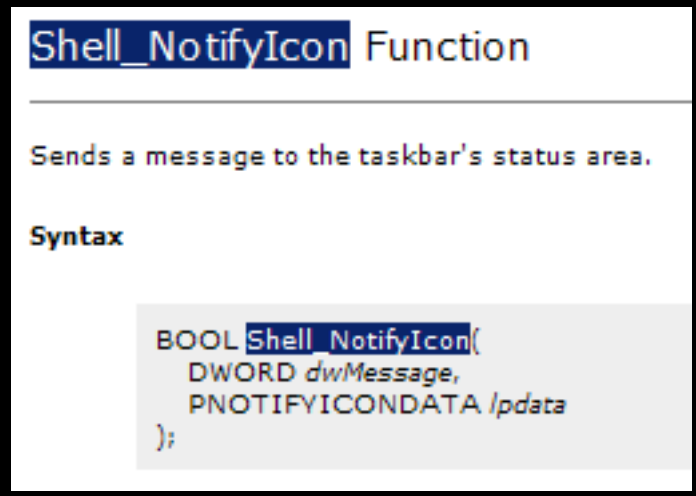
Problem Number #2

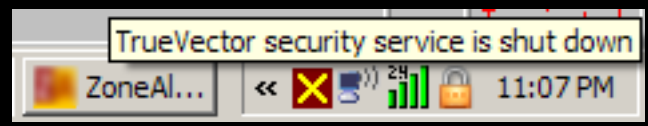
- Various alerts pop up
  - Icon changes
  - Text box
  - Hovering box over text
- Need to get rid of these



Solution to Icon Change

- Calls to Shell_NotifyIcon
- Two parameters
- add esp, 8
- We win.





Screen Text

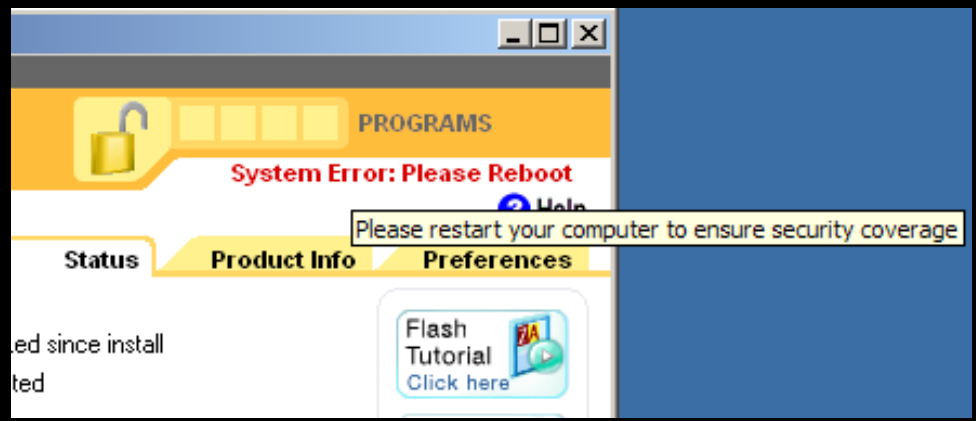
- Uses a few resource strings
  - In .rsrc section of PE
- Overwrite them!
  - With the originals
- Uses DrawText to write to screen

DrawText

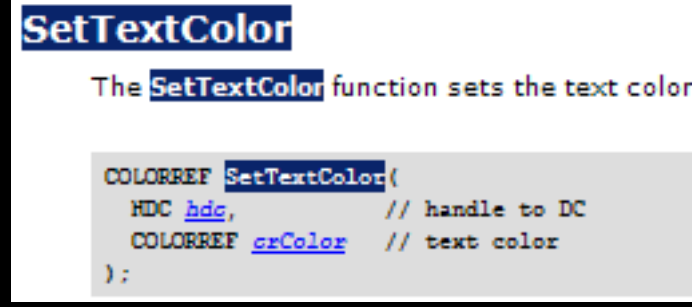
The **DrawText** function draws formatted text

To specify additional formatting options, use

```
int DrawText(
  HDC hDC,           // handle to DC
  LPCTSTR lpString,  // text to draw
  int nCount,        // text length
  LPRECT lpRect,     // formatting dimensions
  UINT uFormat       // text-drawing options
);
```



Solution!



- User-mode hook on DrawText
- Need to preserve registers:
  - Pusha, pushfd, popfd, popa
- Mov ebx, [esp+2c] after pushad pushfd
- Check [ebx] for “All ”
- If its there, run SetTextColor to set text to green
- Do the rest of the stack setup, and we win.

Transporting the Driver

- Include it as a resource in a dll
- DLL injection code can load the rest
- Use FindResource(), LoadResource(), LockResource(), WriteFile() to write to disk
- OpenSCManager(), CreateService(), StartService() to load the driver

Coercing Execution

- Use WriteProcessMemory() to write the code to the addresses
- The shutdown code will be written to empty space in the executable
- CreateRemoteThread() called to execute the code
- We do this after we've changed all the strings and injected the hooks

Demo

Conclusion

- Things to be learned
  - Firewalls are just applications
  - They are not the end all of security
- Thanks to: HD Moore and Optyx