Bypassing Secure Desktops Protections

Bruno Oliveira & Márcio Almeida
Agenda

① Who Are We?
② Intro To Secure Desktop
   - What is it?
   - What does it work?
③ Windows API
④ Our PoC
⑤ Mitigation
⑥ Conclusions
Who are we?

Don’t know you

• Bruno Gonçalves de Oliveira
  – Senior SpiderLabs Security Consultant
  – MSc Candidate
  – Offensive Security
  – Talks at AppSec USA 14, THOTCON, SOURCE Boston, Black Hat DC, SOURCE Barcelona, DEF CON, Hack In The Box, ToorCon, Ekoparty, YSTS & H2HC.

• Márcio Almeida Macêdo
  – SpiderLabs Security Consultant
  – MSc Degree focusing in Web Applications Security – UFPE
Secure Desktop

What is it?

• A way to protect against keystrokes sniffers.

• A new desktop created from the *original* one that should isolate the application.

• Only accessed with SYSTEM privileges.

---

**Enter Master Key on Secure Desktop (Protection against Keyloggers)**

*Note: KeePass was one of the first (maybe even the first) password manager that allows entering the master key on a secure desktop!*

KeePass 2.x has an option (in 'Tools' -> 'Options' -> tab 'Security') to show the master key dialog on a secure desktop (supported on Windows ≥ 2000), similar to Windows' User Account Control (UAC). Almost no keylogger works on a secure desktop.

The option is disabled by default for compatibility reasons.

**KeePass 2.x Only**

*Note that auto-type can be secured against keyloggers, too, by using Two-Channel Auto-Type Obfuscation.*
Secure Desktop

How does it work?

• It is utilized the functions from Desktop objects (Windows API) to create the new desktop.

• It is only accessed with SYSTEM privileges.
Demo 1
How SD works?
Demo 2
Injecting payload on process
Demo 3
Courtesy Shell – VNC Payload
Windows API
Desktop Functions (user32.dll)

- CloseDesktop
- CreateDesktop
- EnumDesktops
- GetThreadDesktop
- OpenDesktop
- OpenInputDesktop
- SetThreadDesktop
- SwitchDesktop

Sessions, Windows Stations and Desktops

Windows API

- Session 0
- Windows Station
- Desktop
- Another Application
- Application
What the Applications do?  
Windows API
Our Attack

Windows API

Session 0

Windows Station

Desktop

Another Application

Application

Secure Desktop

KeyLogger Process

User Input

...
Attack Details

Proof-Of-Concept

- Utilizing OpenDesktop (user32.dll) function request the desktop to be opened.
- Utilizing SetThreadDesktop (user32.dll) get access to desktop.
- Utilizing CreateProcess (kernel32.dll) Start a KeyLogger process into this desktop.
- Get the user input via the KeyLogger process into the “Secured Desktop”.

Trustwave®
SpiderLabs®
static void Main(string[] args) {
    IntPtr hNewDesktop;
    while (true) {
        foreach (string desktop in GetDesktops()) {
            if (!hasP0wn3d(desktop)) {
                hNewDesktop = Open(desktop);
                Task.Factory.StartNew(() => {
                    SetThreadDesktop(hNewDesktop);
                    CreateProcess("c:\windows\system32\cmd.exe", desktop);
                }).Wait();
                _p0wn3d_desktops.Add(desktop);
            }
        }
    }
}
DEMO 4
Proof of Concept
Session Isolation

Windows API
Solution Adopted by 1Password (CVE-2014-3753)
Solution Adopted by 1Password

CVE-2014-3753

Detect if the 1Password is the unique process/program running into the Secure Desktop and if isn’t close the desktop and alert the user.
Conclusions
THANK YOU