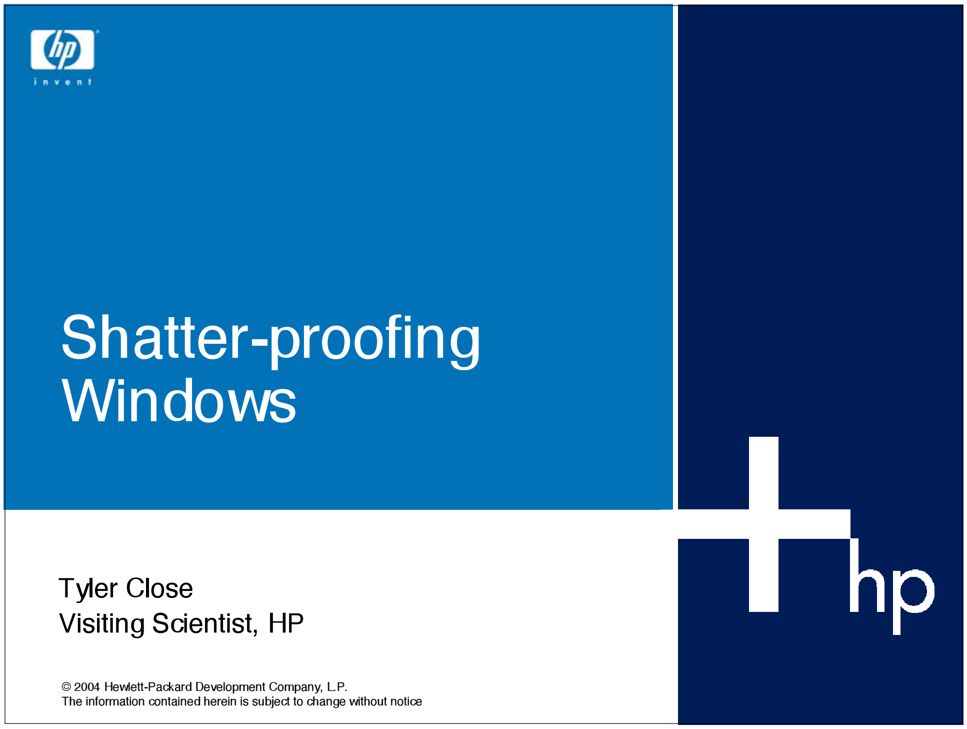
Tyler Close

Shatter-proofing Windows

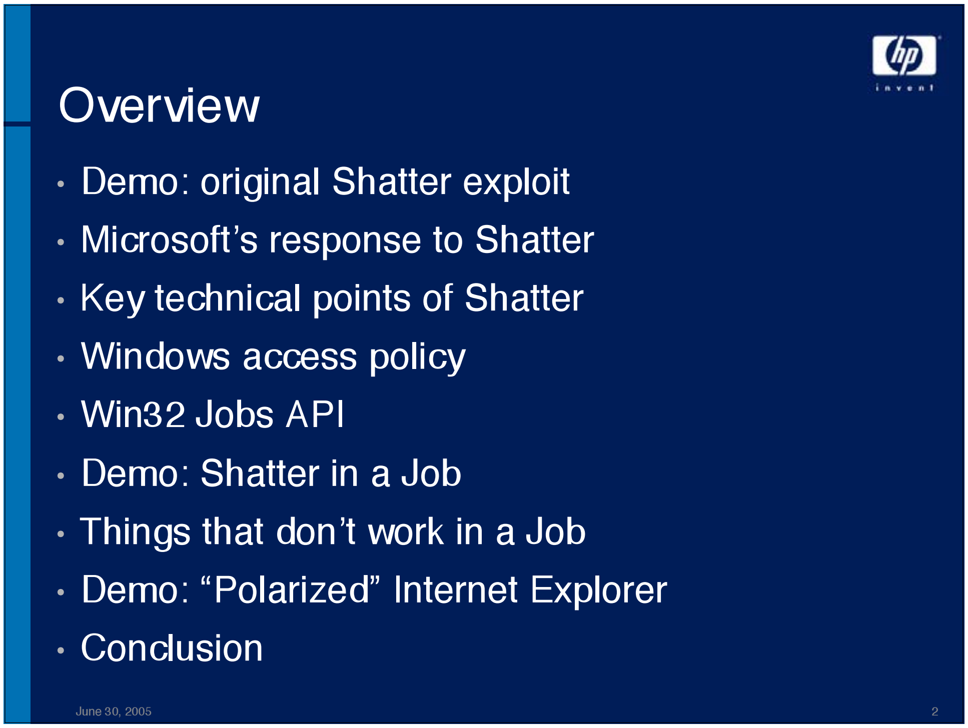
The Shatter attack uses the Windows API to subvert processes running with greater privilege than the attack code. The author of the Shatter code has made strong claims about the difficulty of fixing the underlying problem, while Microsoft has, with one exception, claimed that the attack isn't a problem at all. Whether or not Shatter is indeed an exploit worth worrying about, it uses a feature of Windows that has other malicious uses, such as keystroke logging. This talk presents a means of defeating this entire family of attacks with minimal breaking of applications and effect on the look and feel of the user interface.

Tyler Close *is a researcher and developer, working in the field of secure, multi-user, distributed applications since 1998. He is the designer of the web-calculus, a messaging model for creating POLA interfaces between heterogeneous applications. He is a developer for an ongoing series of applications in the POLA genre, including: Waterken Server, for web-services; petname tool, anti-phishing browser extension; httpsy, decentralized authentication for the WWW; E language, P2P scripting language; Waterken DB, capability-based object database; Waterken IOU, generic rights transfer protocol. Tyler joined HP as a Visiting Scientist in 2005 to work on the Virus Safe Computing Initiative.*



Shatter-proofing Windows

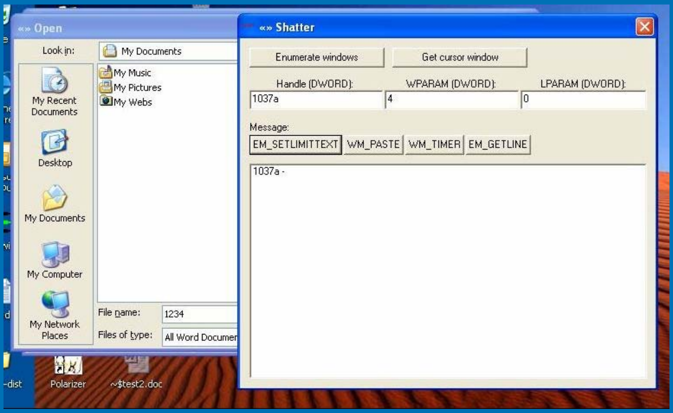
Tyler Close
Visiting Scientist, HP



Overview

- Demo: original Shatter exploit
- Microsoft’s response to Shatter
- Key technical points of Shatter
- Windows access policy
- Win32 Jobs API
- Demo: Shatter in a Job
- Things that don’t work in a Job
- Demo: “Polarized” Internet Explorer
- Conclusion

Demo: original Shatter exploit



File name: 1234

Files of type: All Word Documer

Handle (DWORD): 1037a

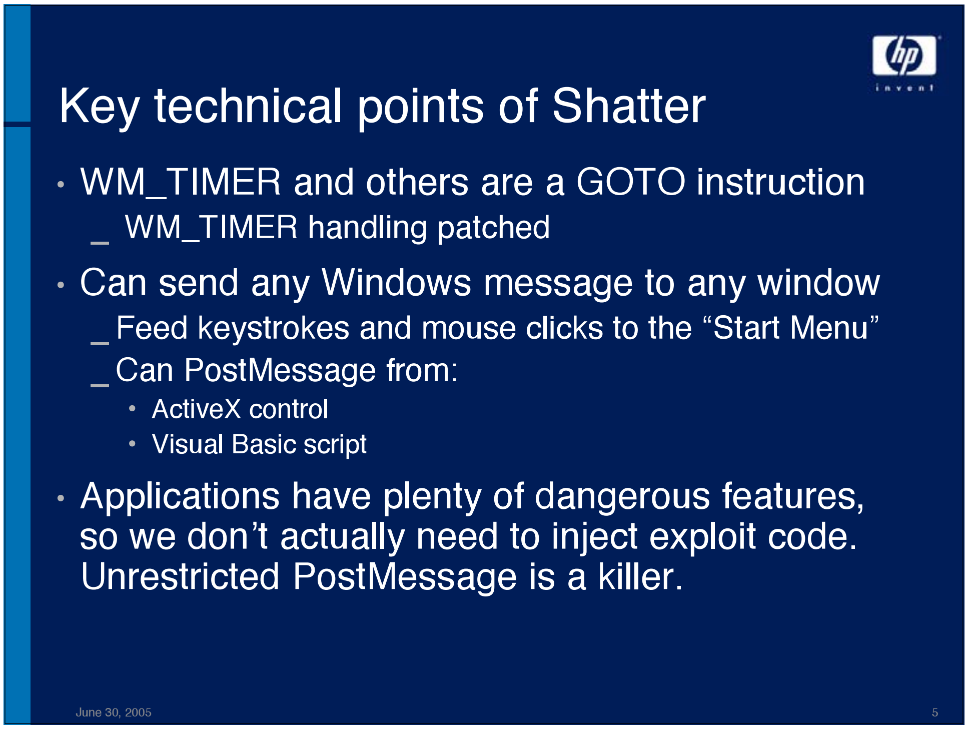
WPARAM (DWORD): 4

LPARAM (DWORD): 0

1037a -

> “By design, all services within the interactive desktop are peers, and can levy requests upon each other.”

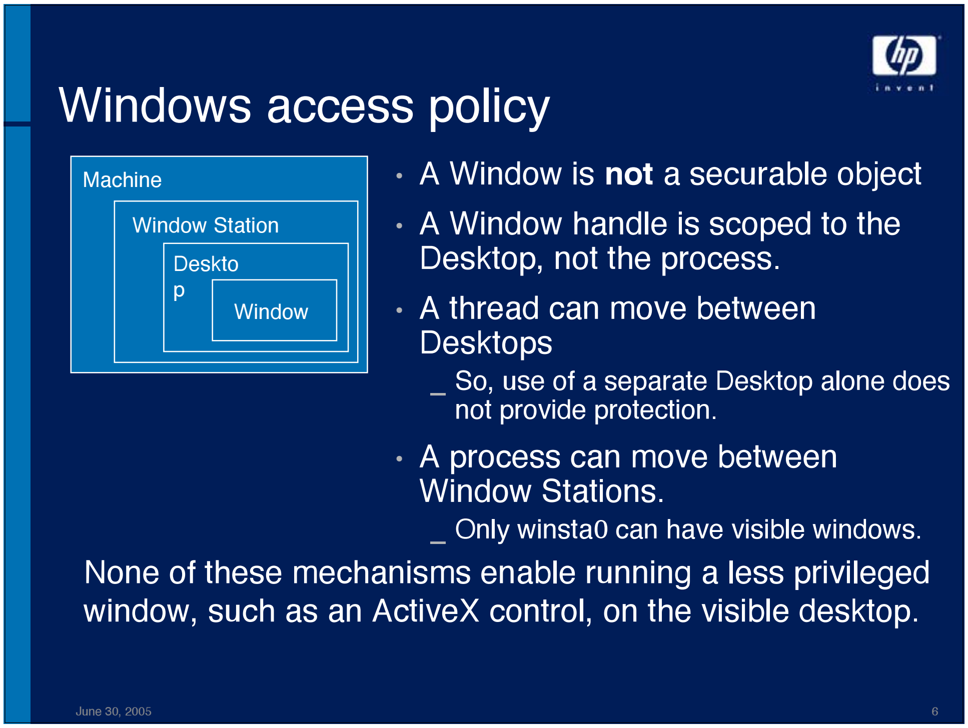
TechNet
Microsoft



Key technical points of Shatter

- WM_TIMER and others are a GOTO instruction
  - _ WM_TIMER handling patched
- Can send any Windows message to any window
  - _ Feed keystrokes and mouse clicks to the “Start Menu”
  - _ Can PostMessage from:
    - ActiveX control
    - Visual Basic script
- Applications have plenty of dangerous features, so we don’t actually need to inject exploit code. Unrestricted PostMessage is a killer.

June 30, 2005 5



Windows access policy

Machine

Window Station

Deskto
p

Window

- A Window is **not** a securable object
- A Window handle is scoped to the Desktop, not the process.
- A thread can move between Desktops
  - _ So, use of a separate Desktop alone does not provide protection.
- A process can move between Window Stations.
  - _ Only winsta0 can have visible windows.

None of these mechanisms enable running a less privileged window, such as an ActiveX control, on the visible desktop.

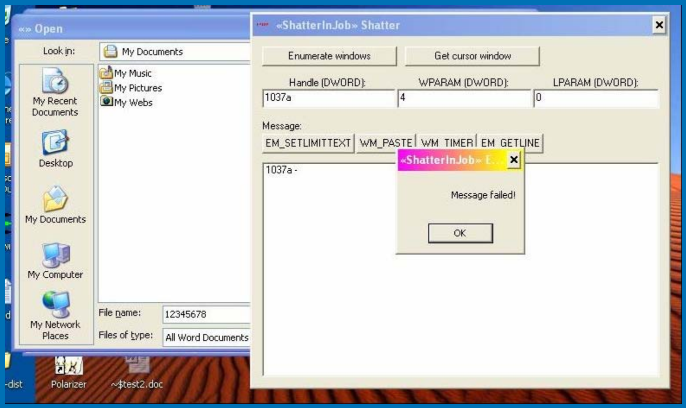
June 30, 2005 6

Win32 Jobs API

- A process assigned to a Job with the UILIMIT_HANDLES restriction cannot use a handle owned by a process outside the job.
- A process cannot escape the Job.

```
char* cmd = "cmd.exe";
HANDLE job = CreateJobObject(NULL, NULL);
JOBOBJECT_BASIC_UI_RESTRICTIONS buir = { JOB_OBJECT_UILIMIT_HANDLES };
SetInformationJobObject(job, JobObjectBasicUIRestrictions, &buir,
                        sizeof buir);
UserHandleGrantAccess(GetDesktopWindow(), job, TRUE);
STARTUPINFO si = { sizeof(STARTUPINFO) };
PROCESS_INFORMATION pi = {};
CreateProcess(NULL, cmd, NULL, NULL, FALSE, CREATE_SUSPENDED, NULL,
              NULL, &si, &pi);
AssignProcessToJobObject(job, pi.hProcess);
ResumeThread(pi.hThread);
```

Demo: Shatter in a Job



File name: 12345678

Files of type: All Word Documents

Handle (DWORD): 1037a

WPARAM (DWORD): 4

LPARAM (DWORD): 0

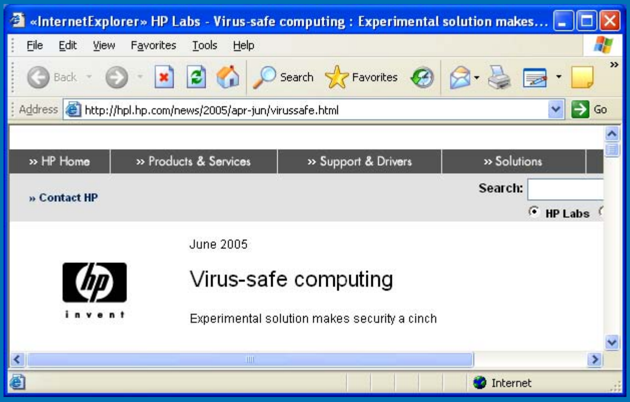
1037a -

Message failed!

Things that don’t work inside a job

- GetCliboardData() for text
  - _ But not for anything else!
- Some apps won’t launch without access to the Desktop window
- Occasional problems with update of mouse cursor
- Drag and drop from an application inside the Job to one outside the Job.
- The Polaris project is aimed at working around these, and other, deficiencies.

Demo: “Polarized” Internet Explorer



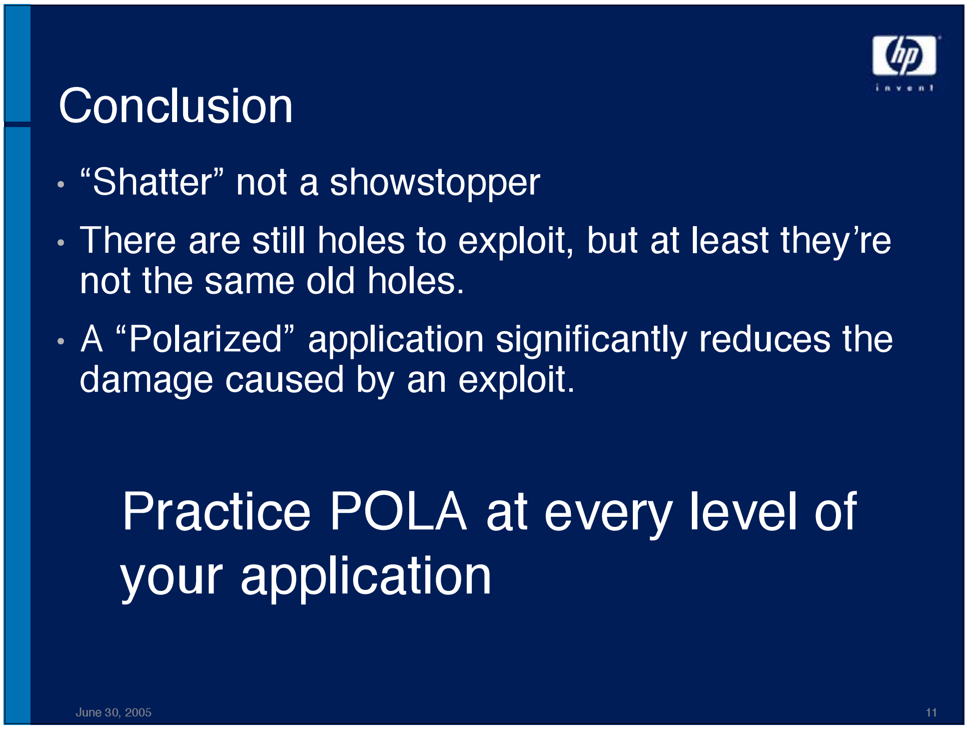
«InternetExplorer» HP Labs - Virus-safe computing : Experimental solution makes...

Address http://hpl.hp.com/news/2005/apr-jun/virussafe.html

June 2005

Virus-safe computing

Experimental solution makes security a cinch



Conclusion

- “Shatter” not a showstopper
- There are still holes to exploit, but at least they’re not the same old holes.
- A “Polarized” application significantly reduces the damage caused by an exploit.

Practice POLA at every level of your application

June 30, 2005 11