Dangling Pointer

Jonathan Afek, 2/8/07, BlackHat USA

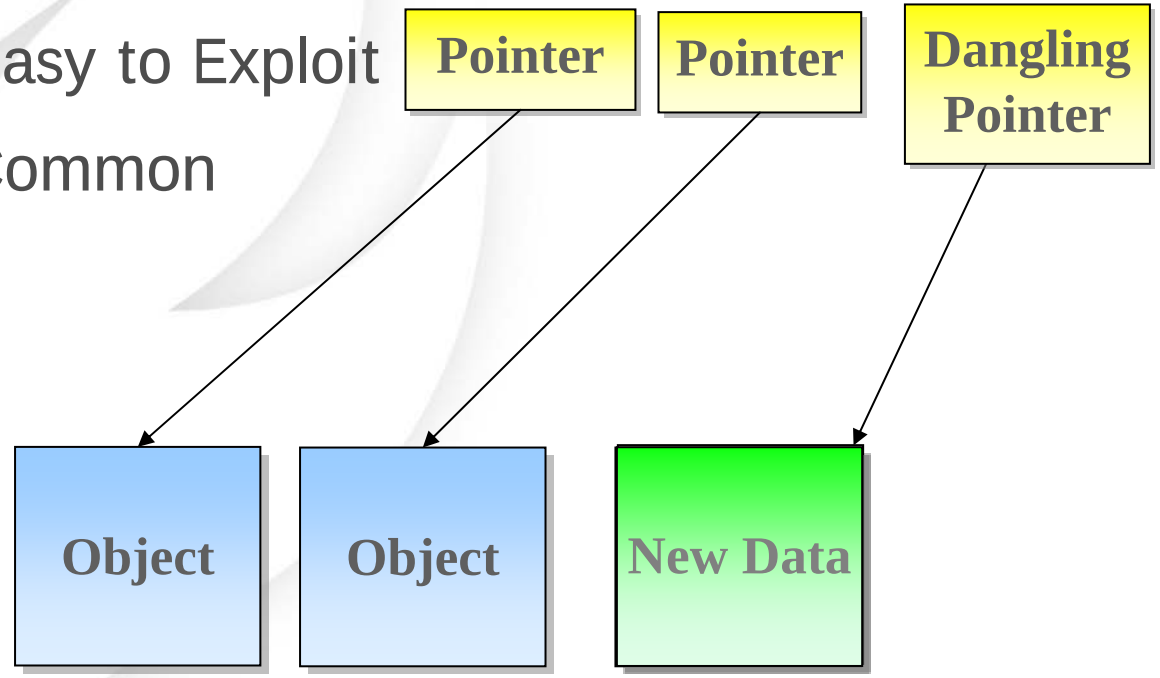
Table of Contents

watchfire

What is a Dangling Pointer?

Invalid Pointer:

- Dangerous
- Easy to Exploit
- Common

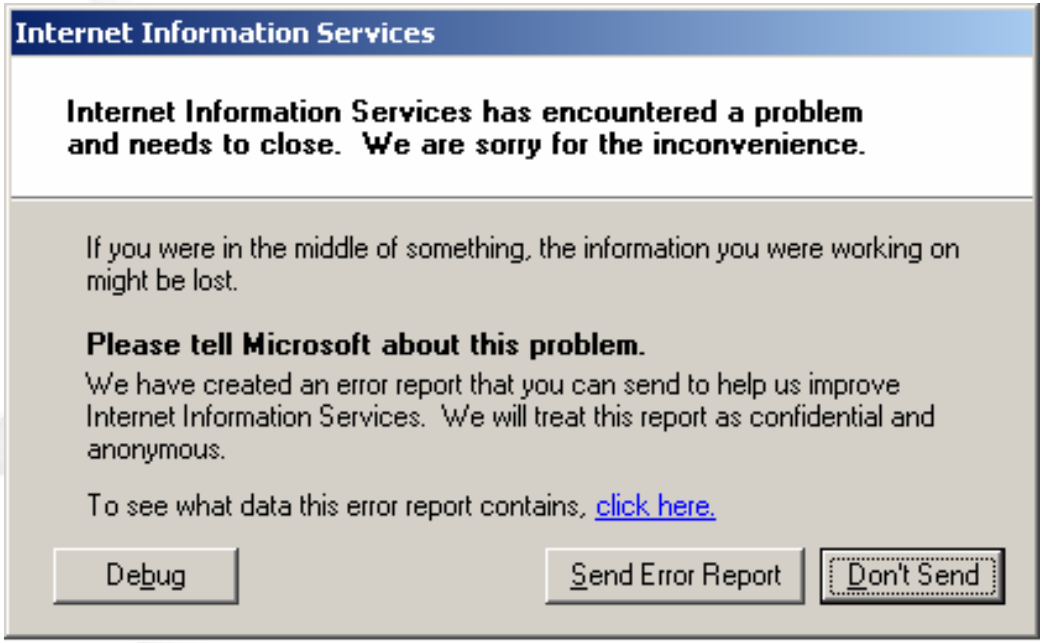


watchfire

What is a Dangling Pointer? – An Example

- Results:

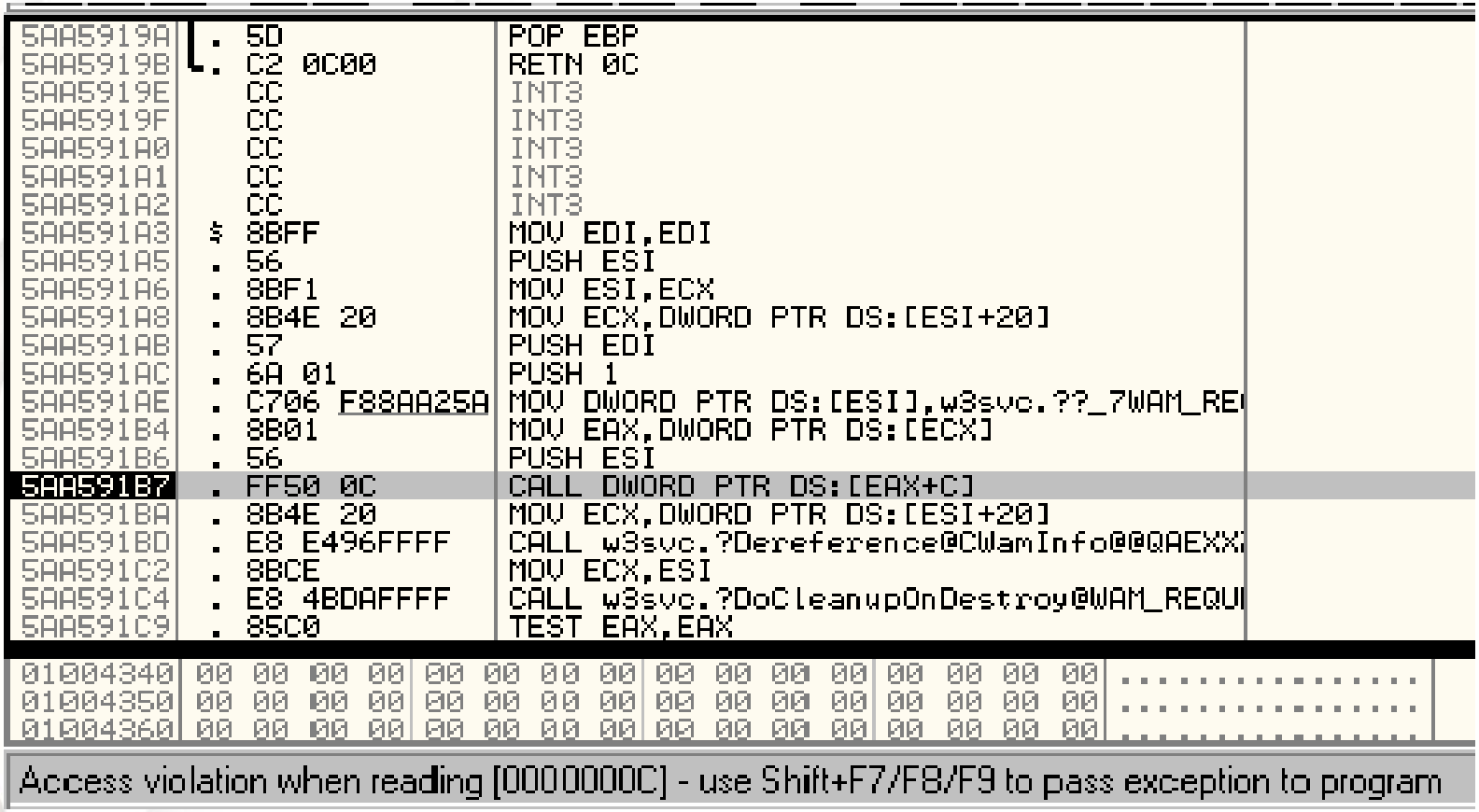
Crash



watchfire

What is a Dangling Pointer? – An Example

- Debugger View



watchfire

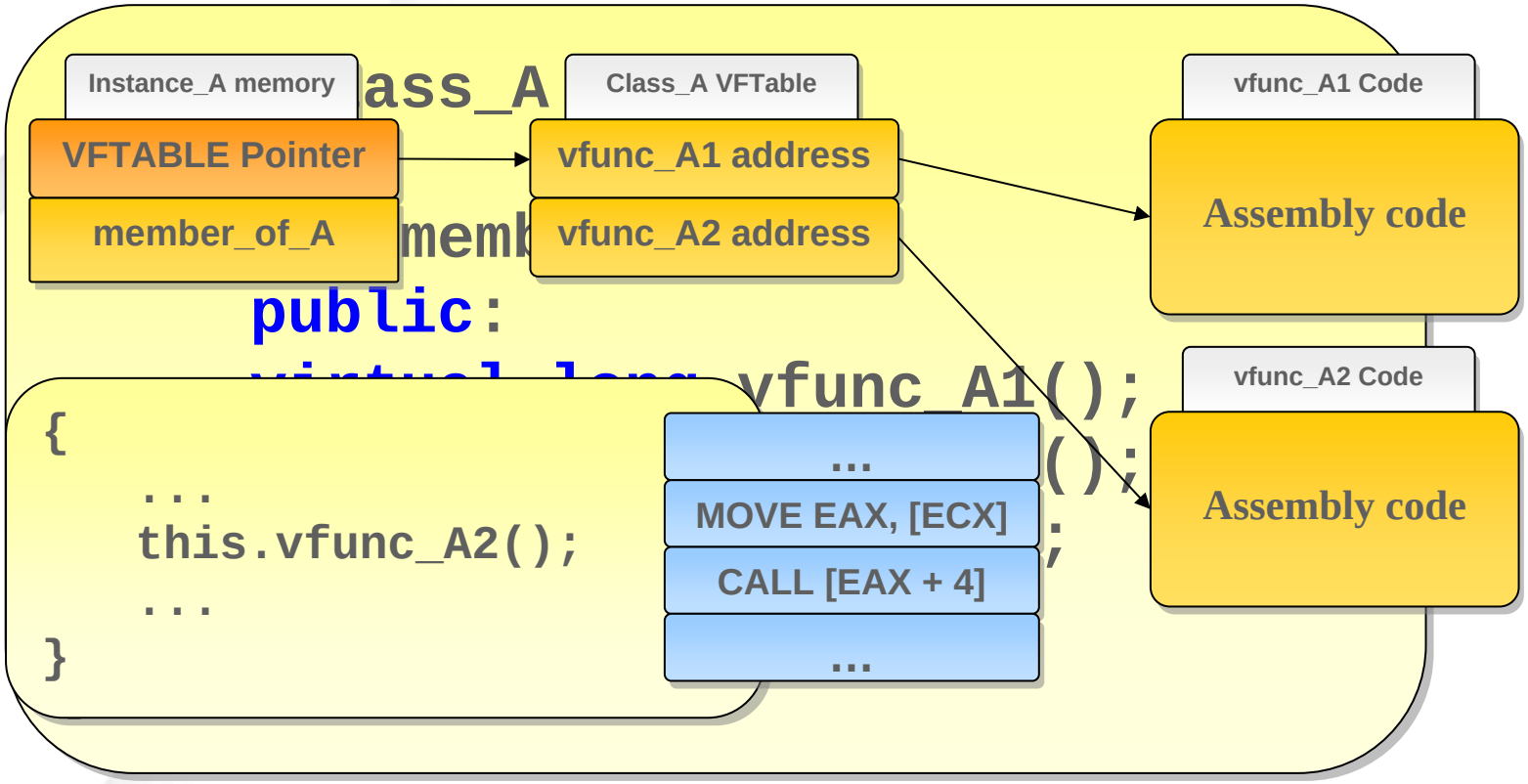
Where are We

- What is a Dangling Pointer?
- **Code Injection**
- Object Overriding
- Demonstrations
- Remediation
- Summary
- Q&A

watchfire

Code Injection – The Layout of an Object

- Class_A:



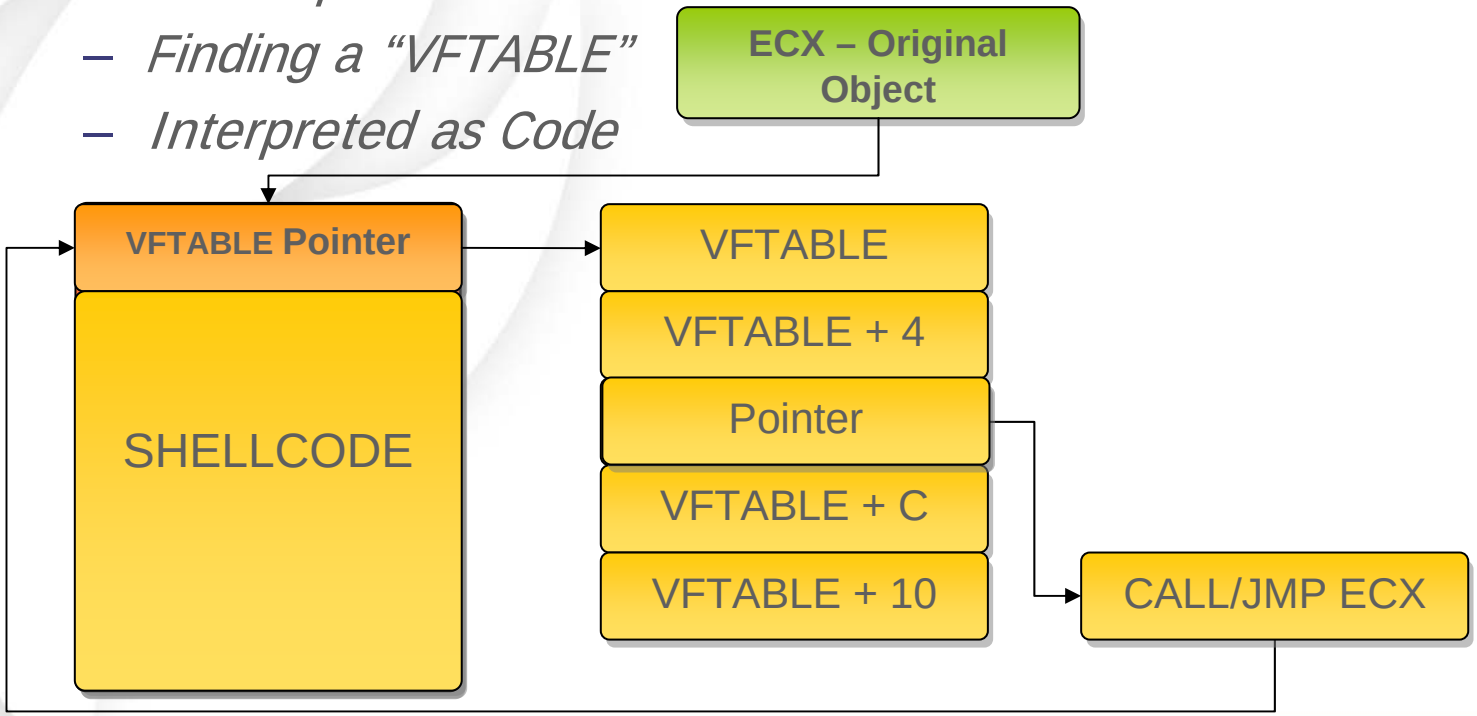
Code Injection – The Double Reference Exploit

Exploit Overview:

- Free the Object
- Override the Object – covered later
- Execute a Virtual Function

Code Injection – The Double Reference Exploit

- *Injecting Code*
  - *Free the Object*
  - *Shellcode*
  - *Call/Jmp ECX*
  - *Finding a "VFTABLE"*
  - *Interpreted as Code*
- *Continue*
  - *Automation*

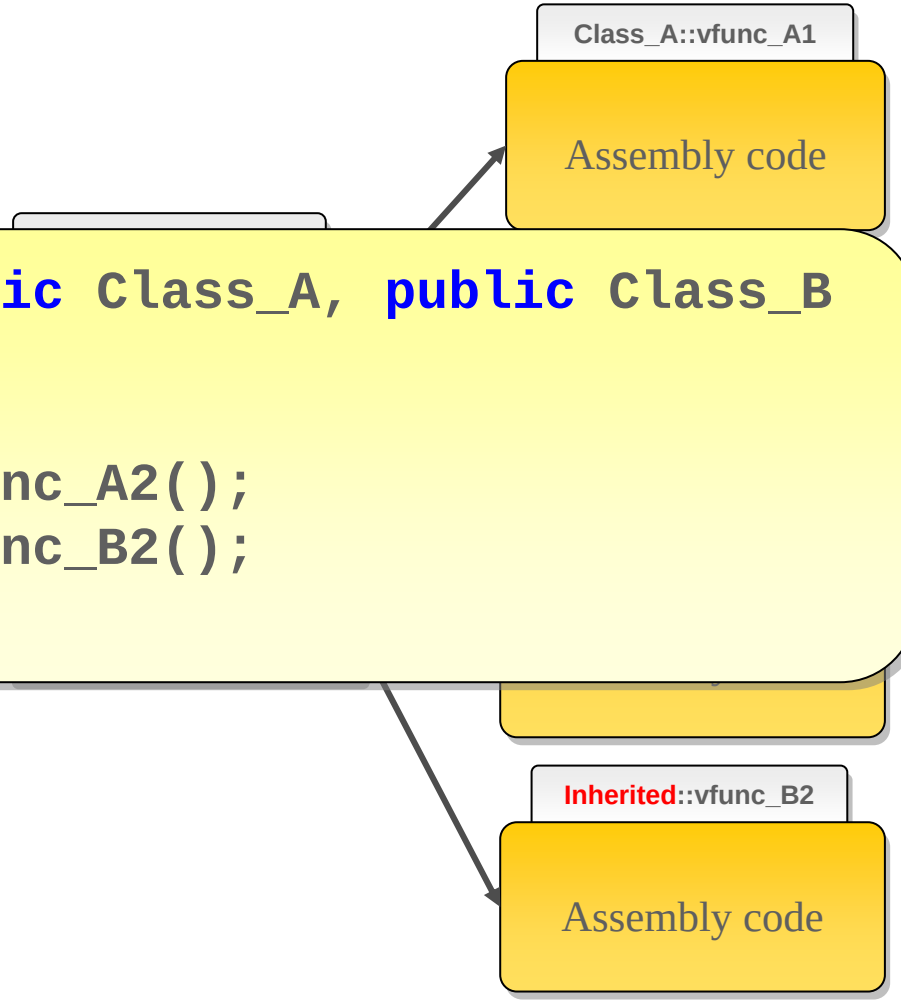


watchfire

Code Injection – Double Inheritance

- Multiple Inheritance

```
class Inherited: public Class_A, public Class_B
{
public:
      virtual int vfunc_A2();
      virtual int vfunc_B2();
};
```



Where are We

- What is a Dangling Pointer?
- Code Injection
- **Object Overriding**
- Demonstrations
- Remediation
- Summary
- Q&A

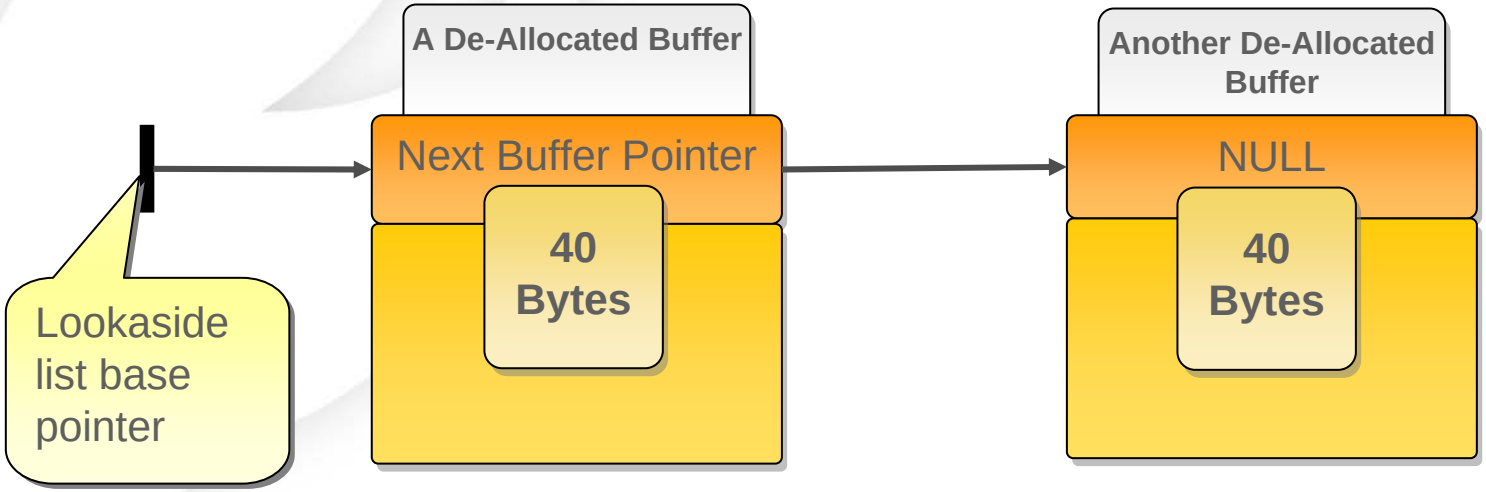
watchfire

Object Overriding

- Allocation Implementation
  - Numerous heaps
    - Two Default heaps
    - Different API
    - C-Runtime functions
      - Malloc
      - Free
      - New
      - Delete
      - Etc.

Object Overriding

- Allocation implementation details
  - Lookaside List
    - A list for each size (8-1024) (8) and for each heap
    - First Allocation Priority
    - Merges



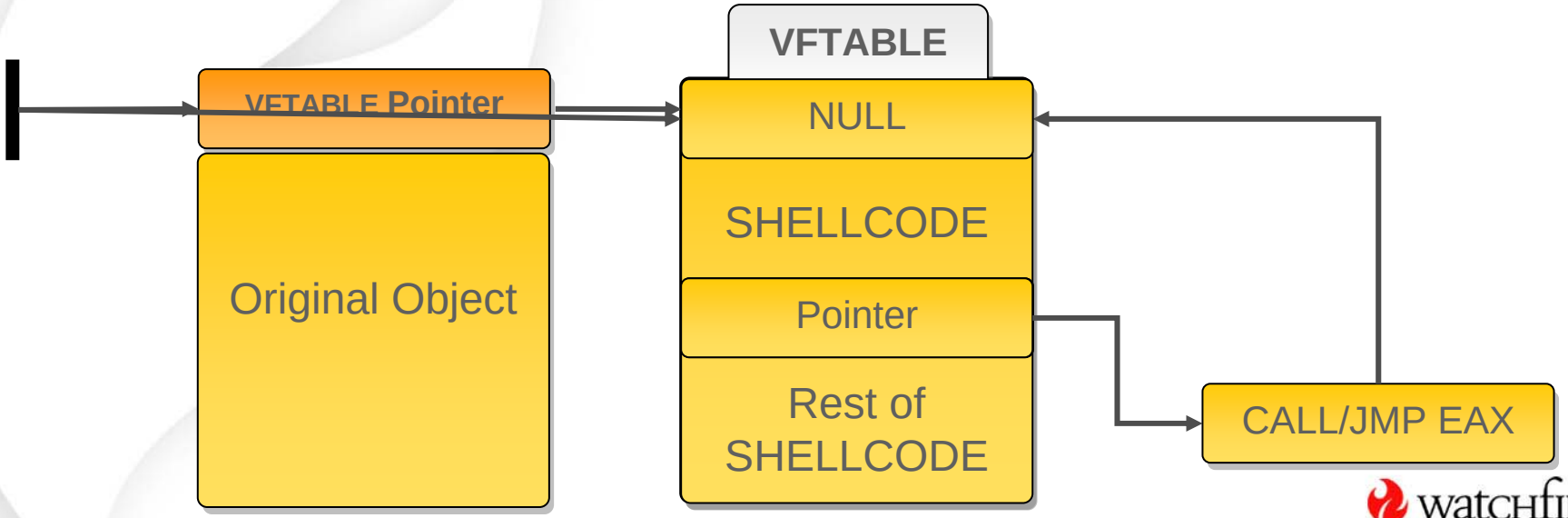
watchfire

Object Overriding

- And Finally – Overriding
  - Search for Allocations
    - Static Analysis
      - Method: Disassembly
      - Restriction: Static Size
      - Validation: Controllable Content
      - Usage: Causing the Allocation
    - Dynamic analysis
      - Method: API Breakpoints
      - Restriction: Static/Dynamic Size
      - Validation: Controllable Content

Object Overriding – The VFTABLE Exploit

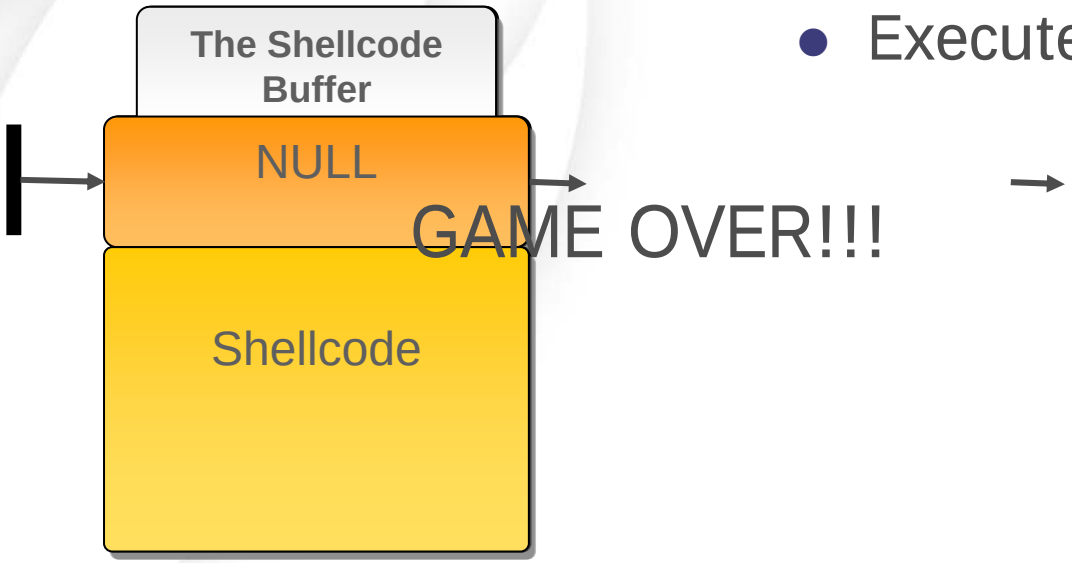
- Exploitation:
  - Empty the Lookaside List
  - Allocate a Buffer
  - Insert Content
  - Free the Buffer
- Continue:
  - Free the Object
  - Execute a VFunc



watchfire

Object Overriding – The Lookaside Exploit

- Empty the Lookaside
- Allocate Two Buffers
- Insert Shellcode



- Free One Buffer
- Free The Other
- Free The Object
- Execute the Destructor

Object Overriding – The Lookaside Exploit

- Executing NULL – NO Problem

watchfire

Summary

- Summary
  - Double Reference
    - Controllable First DWORD
    - Static Address
  - VFTABLE Exploit
    - Controllable Allocations
    - No First DWORD
    - Static Address
  - Lookaside Exploit
    - Controllable Allocations
    - No First DWORD
    - No Static Address
    - Destructor Execution

watchfire

Where are We

- What is a Dangling Pointer?
- Code Injection
- Object Overriding
- **Demonstrations**
- Remediation
- Summary
- Q&A

watchfire

Demonstrations – Configuration Item

- Allocating the Object
- De-Allocation the Object

Demonstrations – Configuration Item

- Allocating User Data

Demonstrations – Configuration Item

- Executing a VFunc

Demonstrations – Configuration Item

- Putting it Together
  - De-Allocate
  - Re-Allocate
  - Execute

watchfire

Demonstrations – Remote Exploit

- Another Exploit on IIS,
  but this time – a remote one

watchfire

Where are We

- What is a Dangling Pointer
- Code Injection
- Object Overriding
- Demonstrations
- **Remediation?**
- Summary
- Q&A

watchfire

Remediation

- Known Protection Mechanisms
  - NX Bit
  - ASLR
- VFTABLE Sanitation
- Safe Programming

watchfire

Summary

- Technical Background
  - Memory Allocations
  - Objects Implementation
- Exploits
  - Double Reference Exploit
  - VFTABLE Exploit
  - Lookaside Exploit
- Demonstrations
  - Configuration Item
  - Remote Exploit
- Dangling Pointer
  - Only Object Oriented Objects

watchfire

Questions

- Ask Away...

watchfire