before we start

my dedication to OWASP :)

I returned 3 days earlier from Portugal to participate on London Chapter event

... kids were not impressed (photo before boarding plane 5 hours ago) ...
OBJECTIVE OF TODAY’S SESSION
WHAT AM I DOING HERE?

• I’m making the business case for you to:

  focus,

  invest time & resources,

  use,

  contribute and maybe even

  sponsor

the **OWASP O2 Platform** project
WHAT IS O2?

and the OWASP O2 PLATFORM
is an: OPEN PLATFORM.
for

AUTOMATING.
APPLICATION SECURITY.
and

WORKFLOWS.
is an:

OPEN PLATFORM
for
AUTOMATING
APPLICATION SECURITY
KNOWLEDGE
and
WORKFLOWS

Saturday, 5 September 2009
... and when you start using it ...

... you will be able to do impossible things ...
and your clients will love you
‘Joined trace’ example (before):

Example of two separate Traces of an HacmeBank Web Service call (vulnerable to SQL Injection)

Saturday, 5 September 2009
‘Joined trace’ example (after):

Example of a single ‘Joined Trace’ of the same HacmeBank Web Service call (vulnerable to SQL Injection)
TECHNOLOGIES SUPPORTED by O2
Supported Technologies

- **Ounce Labs Scanner:** (FULL Support): scanning, CIR consumption, rules creation, open & save findings format. Languages: .NET, Java, C/C++, ASP Classic, VB 6.0
- **IBM AppScan Developer Edition:** open findings format. Language: Java
- **Microsoft CAT.NET scanner:** scanning, open findings format. Language: .NET (C#, VB.net, Iron Phyton, etc...)
- **FindBugs scanner:** open findings format. Language: Java
- **OWASP CodeCrawler:** open findings format. Language: .NET
- **Fortify (very early stages):** open findings format (FVDL). Language: .NET, Java, C/C++, etc..
- **.NET** - create CIR, create call flow traces, create run-time traces
- **Java** - create CIR, create call flow traces
- **Spring MVC ‘Annotation Based Controllers’** - Model controllers behavior, drive BlackBox tests
So what can O2 do for Advanced Users

- This is for users who know what they (technically) want to do
- These are the users that ALL/MOST tool vendors don't cater for today (since they are not a big enough market)

- The following (are some of the) problems that O2 has solutions for:
  - Advanced findings filtering (for example query 50M to 500Mb assessment files)
  - Visualizing traces
  - Mass rule creation & management
  - “Rules Driven Scans”
  - Creating ALL Traces
  - Joining and Manipulating Traces
  - Scripting questions and workflows (on top of rich objects like CirData, Findings or Rules)
  - Gain visibility into Frameworks
  - Understand and exploit Spring MVC apps
  - Integrate complex workflows with SDLs
  - Do Virtual Patching
  - Quickly Write PoCs and exploits using O2’s .NET’s power Debugger
  - Create “Run-time traces”
  - Write Unit Tests for PoCs
  - Find (via instrumenting and automating the security consultant’s brain) all sorts of application security issues (like to ones in the OWASP Top 10)
  - Start venturing into Source-Code-Fixing for vulnerabilities found
  - Start venturing into auto-writing WAF rules for vulnerabilities found
O2 MODULES
O2 MODULES - DEVELOPMENT STATE

ACTIVE

- O2 Tool - Findings Viewer
- O2 Tool - CirViewer
- O2 Tool - Rules Manager
- O2 Cmd - Findings Filter
- O2 Cmd - Spring MVC
- O2 Tool - Join Traces
- O2 Debugger Mdbg
- O2 Tool - CSharpScripts
- O2 Scanner - MsCatNet
- O2 Tool - Host Local Website
- O2 Tool - Java Execution
- O2 Tool - O2 Scripts
- O2 Tool - Python
- O2 Tool - Search Engine

LEGACY

- O2 Scanners
- O2 Tool - DotNet Callbacks Maker
- O2 Tool - Findings Query
- O2 Tool - Search Assessment Run
- O2 Tool - View Assessment Run
- O2 Tool - WebInspect Converter

Vaporware

- O2 Tool - Filter Assessment Files
- O2 Tool - O2 Reflector

Saturday, 5 September 2009
This graph maps the module’s features to the current version of Ounce’s 6.x OSA (Ounce Security Analyst).

- O2 Tool - Rules Manager
- O2 Tool - CirViewer
- O2 Cmd - Spring MVC
- O2 Debugger Mdbg
- O2 Tool - CSharpScripts
- O2 Scanner - MsCatNet
- O2 Tool - Host Local Website
- O2 Tool - Java Execution
- O2 Tool - Join Traces
- O2 Tool - O2 Reflector
- O2 Tool - O2 Scripts
- O2 Tool - Python
- O2 Tool - WebInspect Converter

- O2 Tool - Findings Viewer
- O2 Cmd - Findings Filter
- O2 Tool - Findings Query
- O2 Tool - DotNet Callbacks Maker
- O2 Tool - Filter Assessment Files
- O2 Tool - Search Engine
- O2 Tool - Search Assessment Run
- O2 Tool - View Assessment Run

- O2 Scanners
O2 MODULES - COMMERCIAL SOFTWARE DEPENDENCIES

COMMERCIAL TOOLS ($$$)

- O2 Tool - Web Inspect Converter
- O2 Tool - DotNet Callbacks Maker
- O2 Tool - Filter Assessment Files
- O2 Scanners

OPEN SOURCE (or FREE)

- O2 Cmd - Spring MVC
- O2 Cmd - Findings Filter
- O2 Tool - O2 Reflector
- O2 Tool - O2 Scripts
- O2 Tool - Java Execution

- O2 Tool - O2 Python
- O2 Tool - O2 Scans
- O2 Tool - C-Sharp Scripts
- O2 Tool - Cir Viewer
- O2 Scanner - MsCatNet

- O2 Tool - Host Local Website
- O2 Tool - Search Engine
- O2 Tool - Search Assessment Run
- O2 Tool - Findings Viewer
- O2 Tool - Findings Query
- O2 Tool - O2 Reflector
- O2 Tool - O2 Scripts
- O2 Tool - Java Execution

Saturday, 5 September 2009
O2 MODULES DETAILS
ACTIVE O2 MODULES (6X)

see presentation
for individual slides with O2 Modules details
ACTIVE O2 MODULES (6X)

O2 Cmd - Findings Filter

DESCRIPTION
This O2 module shows how command line tools can be easily created to provide specific functionality (based on business requirements) which can then be easily integrated on an IDE.

This module takes advantage of the highly flexible O2 Findings Object model.

A GUI is provided to execute and customize the implemented filters.

KEY / UNIQUE FEATURES
- Filters implemented on onlyTraces, noTraces, allFindings, onlyHigh, onlyMedium, onlyLow, onlyInfo, onlyNotes, noneTraces.
- Ability to create assessment files that can be published.
- Ability to dynamically compile and execute custom filters.

USE CASES
- SOX Integration
- Custom Findings Filtering

PRODUCTIZATION READINESS

CAN OSA DO THIS?

Yes

COMMERCIAL SOFTWARE DEPENDENCIES:
Consumes .o2 files created by Ounce (or others).

ROADMAP: NEXT DEVELOPMENT
- Add support for the other ‘O2 Supported’ Scripting languages (ie:Python, Perl, etc)
- Add more sample scripts.
- Document functionality and O2’s command line tools architecture.
- Fix bug with ‘Unique Traces’ filter (as reported by Eduarda).
- Run on Mono (Linux and Mac).

see presentation
for individual slides with O2 Modules details
ACTIVE O2 MODULES (6X)

ACTIVE
MODULE:


for individual slides with O2 Modules details
ACTIVE O2 MODULES (6X)

O2 Cmd - Findings Filter
DESCRIPTION
This O2 module shows how command line tools can be easily created in runtime.

O2 Cmd - Spring MVC
DESCRIPTION
Specifically targeted at the Java Spring MVC framework, this module is able to

O2 Debugger Mdbg
DESCRIPTION
Managed wrapper on top of Microsoft's manager debugger.
This module allows the easy debugging of .NET applications (started or hooked into processes).
The power of this module lies on the wrapping of the 'command line' Managed debugger interface into a GUI & a scriptable environment.

KEY / UNIQUE FEATURES
• Ability to 'Animate Tracing' (StepIn StepOut)
• Record traces, Modify values on breakpoints, view object model (via reflection) of running processes
• Issue breakpoint creation
• Allows easy explicit creation and auto-patching of vulnerabilities

USE CASES
• .NET framework debugging
• Vulnerability exploit writing

PRODUCTIZATION READINESS

CAN QA DO THIS?
with reason and insight

PROTOTYPE

COMMERCIAL SOFTWARE DEPENDENCIES:
Free Microsoft Managed Debugger demo application

ROADMAP: NEXT DEVELOPMENT
• Implement the same capabilities for Java and Python
• Add ability to trace into unmanaged traces
• Improve trace creation process and data collection
• Improve patching & hooking workflow

see presentation
for individual slides with O2 Modules details
ACTIVE O2 MODULES (6X)

- **O2 Cmd - Findings Filter**
  - **Description:** The O2 module allows how command line tools can be easily created to remove.

- **O2 Cmd - Spring MVC**
  - **Description:** Specifically targeted at the Java Spring MVC framework, this module is able to...

- **O2 Debugger Mdbg**
  - **Description:** Managed wrapper on top of Microsoft's manager debugger.

- **O2 Scanner - MscAtNet**
  - **Description:** Module that allows the scanning of .NET applications using the freely available Microsoft's .NET scanner with the results being converted into the O2 findings format (XPATH, XML, XSL).
  - **Key / Unique Features:**
    - Ability to scan (recursively) entire directories
    - Ability to convert .NET scanner results into O2 findings schema
    - Uses the Firefox browser to render the webpages shown when the user is asked to install .NET scanner.
  - **Use Cases:** Scan of .NET projects by users with no access to Ounce 6.x

See presentation:


For individual slides with O2 Modules details.
ACTIVE O2 MODULES (6X)

**O2 Cmd - Findings Filter**
*Description*
This tool allows for command-line tools to be easily created in tandem.

**O2 Cmd - Spring MVC**
*Description*
Specifically targeted at the Java Spring MVC framework, this module is able to...

**O2 Debugger Mdbg**
*Description*
Managed wrapper for Java's manager debugger.

**O2 Scanner - MsCatNet**
*Description*
Module that allows the scanning of .NET applications using the freely available...

**O2 Tool - Clr Viewer**
*Description*
This module allows for the creation (.NET) and visualization (all Ounce supported languages) of CIR (O2's version of Ounce's Common Intermediate Representation). By exposing the CIR object model in a powerful GUI (and programmatically object model), this module allows O2 users to gain a much deeper understanding of the application under analysis. It is possible to create O2 Findings from CIR data.

**Key/Unique Features**
- Ability to consume CIR created from all languages supported by the Ounce 6.x engine.
- Ability to create CIR from .NET and Java class files.
- Recursive mapping of function callers, function CallGraphs and SuperClasses/Interfaces.

**Use Cases**
- Visualize an application object model.
- Create call-flow traces (i.e. findings) from CIR.

**Productization Readiness**

**Can O2 do this?**

**Commercial Software Dependencies**
Uses Ounce's internal CIR.

**Roadmap: Next Development**
- Convert O2 ClrGraph representation into the (under development) CIR (Open Intermediate Representation) Scheme.
- Add support for ClrGraph creation for Java class files (functionality already available in the O2 Spring MVC module).

---

See presentation
for individual slides with O2 Modules details.
ACTIVE O2 MODULES (6X)

- **O2 Cmd - Findings Filter**
  - **DESCRIPTION:** This O2 module shows how command line tools can be easily created in O2.

- **O2 Cmd - Spring MVC**
  - **DESCRIPTION:** Specifically designed for the Java Spring MVC framework, this module is able to...

- **O2 Debugger Mdbg**
  - **DESCRIPTION:** Managed wrapper on top of Microsoft’s manager debugger.

- **O2 Scanner - MsCatNet**
  - **DESCRIPTION:** Module that allows the scanning of .NET applications using the freely available...

- **O2 Tool - C# Viewer**
  - **DESCRIPTION:** This module allows the creation (NET) and visualization of all Ounce supported...

- **O2 Tool - C-Sharp Scripts**
  - **DESCRIPTION:** This module is designed to help writing O2 modules in O2. It provides a full compilation and debugging environment (using the same modules as the O2 Debugger Mdbg module) and allows advanced users to write powerful scripts on top of the O2 Object model.
  
  **KEY / UNIQUE FEATURES:**
  - Write analysis scripts (i.e. custom modules) on a managed language (C#)
  - Ability to hook and control the debugging engine (which is how the O2 virtual patching occurs)

  **USE CASES:**
  - Advanced debugging of .NET and ASP.NET applications
  - Exploit development and Virtual patching

  **PRODUCTIZATION READINESS:**
  - CAN OSA DO THIS?
  - YES
  - NO

- **COMMERCIAL SOFTWARE DEPENDENCIES:**
  - Free Microsoft Managed Debugger demo application

- **ROADMAP NEXT DEVELOPMENT**
  - Integrate with the O2 Scripts module in order to create a single scripting environment for O2
  - Loop load required scripting and debugging engines

---

*see presentation*


for individual slides with O2 Modules details

---

Saturday, 5 September 2009
ACTIVE O2 MODULES (6X)

see presentation
for individual slides with O2 Modules details
ACTIVE O2 MODULES (6X)

**O2 Tool - Java Execution**

**DESCRIPTION**
Module that allows users to access the rich O2 object model from Java. Although the code written in Java will seem to have full access to the O2 object model, under the hood Java bytecode can be converted using IKVM into .NET bytecode, which is then able to access and consume the O2 modules used (IKVM has a built-in tool that creates .NET assemblies).

**KEY / UNIQUE FEATURES**
- Wraps IKVM and allows the easy creation of the dependencies required to write O2 Modules in Java (making it easy to script O2 from Eclipse)
- Adds to O2 Soring module the ability to write and execute Java code

**USE CASES**
- Write custom scripts in Java that access or manipulate data stored in O2 Objects

**PRODUCTIZATION READINESS**

**CAN OSA DO THIS?**

Yes

**COMMERCIAL SOFTWARE DEPENDENCIES:**

**ROADMAP: NEXT DEVELOPMENT**
- Further automate the process of using IKVM in O2
- Auto configure Eclipse so that developers can use it (Eclipse) to write and execute their (written) Java code

see presentation


for individual slides with O2 Modules details
ACTIVE O2 MODULES (6X)

O2 Tool - Java Execution

DESCRIPTION
Module that allows users to access the rich O2 object model from Java.

O2 Tool - Join Traces

DESCRIPTION
This module provides a PoC (Proof of Concept) for how traces can be joined together based on simple string matching. Join traces when the Start or End of the trace matches the Trace or End of the trace.x  

KEY / UNIQUE FEATURES
- Join separate traces based on a simple string criteria
- Automatically handle the joining of .NET web services

USE CASES
- Create traces that cross multiple logical or physical boundaries and create highly-actionable findings

PRODUCTIZATION READINESS

CAN OSA DO THIS?

COMMERCIAL SOFTWARE DEPENDENCIES:
- Join traces created by Ounce and others

ROADMAP: NEXT DEVELOPMENT
- Convert join algorithm in new O2 Findings Format
- Add support for more specific trace joins (getters & setters, SetAttributes, HashMap, etc.)

see presentation
for individual slides with O2 Modules details
ACTIVE O2 MODULES (6X)

GEEK-O-METER
manager
analyst
security
consultant
senior
consultant
O2
developer

ACTIVE O2 Tool - Java Execution
DESCRIPTION
Module that allows users to access the rich O2 object model from Java.

ACTIVE & LEGACY O2 Tool - Join Traces
DESCRIPTION
This module provides a PoC (Proof of Concept) for how traces can be joined.

ACTIVE O2 Tool - O2 Scripts
DESCRIPTION
Lightweight O2 scripting environment that supports scripting in .NET, Java, Python, and IronPython.

KEY / UNIQUE FEATURES
- Supports multiple engines
- Exposes O2 object model

USE CASES
- Quickly write Python scripts that consume the O2 engines and access Java jars or .NET assemblies

PRODUCTIZATION READINESS
CAN OSA DO THIS?

see presentation
for individual slides with O2 Modules details
ACTIVE O2 MODULES (6X)

**O2 Tool - Java Execution**
**DESCRIPTION**
Module that allows users to access the rich O2 object model from Java.

**O2 Tool - Join Traces**
**DESCRIPTION**
This module provides a PoC (Proof of Concept) for how traces can be joined.

**O2 Tool - O2 Scripts**
**DESCRIPTION**
Lightweight O2 scripting environment that supports scripting in .NET, Java, or bash.

**O2 Tool - Python**
**DESCRIPTION**
O2 module that allows the quick creation and execution of Python scripts.

**O2 Tool - Rules Manager**
**DESCRIPTION**
Very powerful O2 Module that allows the quick visualization, creation, and editing of Ounce rules. The module supports the O2 Rule Pack format which can be used to import or export rules between different O2 or Ounce computers. This module supports the following common work flow: Scan application, create CQR, create rules from CQR, scan application, create rules from scan findings.

**KEY / UNIQUE FEATURES**
- Import rules from Ounce's MySql database; powerful filtering; create or modify rules; commit changed rules to MySql database; map CQRs and findings to existing rules; automatically create rules from CQRs; apply rules to findings without requiring rescan.

**USE CASES**
- Use of the Ounce Engine
- Use of O2's mini Call-Flow Analysis Engine

**PRODUCTIZATION READINESS**

**CAN OSA DO THIS?**

see presentation
for individual slides with O2 Modules details

Saturday, 5 September 2009
ACTIVE O2 MODULES (6X)

O2 Tool - Java Execution
DESCRIPTION
Module that allows users to access the rich O2 object model from Java.

O2 Tool - Join Traces
DESCRIPTION
This module provides a PoC (Proof of Concept) for how traces can be joined.

O2 Tool - O2 Scripts
DESCRIPTION
Lightweight O2 scripting environment that supports scripting in .NET, Java, Python.

O2 Tool - Python
DESCRIPTION
O2 module that allows the quick creation and execution of Python scripts.

O2 Tool - Rules Manager
DESCRIPTION
Very powerful O2 Module that allows the quick visualization, creation and editing.

O2 Tool - Search Engine
DESCRIPTION
Simple tool to allow the quick Java search of source files (source code, xml, config files, etc.).
During a normal engagement, this tool tends to be used very regularly (from helping to quickly find a particular text string to validating a source code finding).

KEY / UNIQUE FEATURES
- Ability to recursively import files from a drag and dropped directory with quick filtering on file type and display of file size.
- Ability to run multiple searches and to quickly see its source code reference.

USE CASES
- Text search of provided source code artifacts during security assessment.

PRODUCTIZATION READINESS
- CAN OSA DO THIS?
- YES
- NO

COMMERCIAL SOFTWARE DEPENDENCIES:

ROADMAP: NEXT DEVELOPMENT
- Create findings from search results
- Save Search Criteria as a Rule (create rule save format first)
- Allow boolean logic on
- Add 'Search by Proximity' feature

see presentation
for individual slides with O2 Modules details
LEGACY O2 MODULES (6X)

see presentation
for individual slides with O2 Modules details
**LEGACY O2 MODULES (6X)**

**O2 Scanners**

**DESCRIPTION**
O2 module that allows the easy triggering of scans using the Ounce CLI (Command Line Interface) or the Microsoft's CATNET Scanner. This module also contains an earlier version of the O2's CR creation process and a special scanning mode (now discontinued) that aimed at generating all possible traces.

**KEY / UNIQUE FEATURES**
- Standard interface to trigger scans
- Drag & Drop scanning environment
- Ability to run a multi-pass scan on the Ounce 6.x engine that generates all possible traces
- Ability to manually control the CR creation process

**USE CASES**
- Advanced O2 users that want to generate all traces or control the CR creation process

**PRODUCTIZATION READINESS**

**CAN OSA DO THIS?**

**COMMERCIAL SOFTWARE DEPENDENCIES:**
Uses Ounce's and Microsoft's CATNET scanning engine.

**ROADMAP: NEXT DEVELOPMENT**
- This module has been made redundant by the new version of Rule Manager and standalone CATNET scanning module.

---

**see presentation**


for individual slides with O2 Modules details
LEGACY O2 MODULES (6X)

O2 Scanners
DESCRIPTION
O2 module that allows the easy triggering of scans using the Ounce GUI

O2 Tool - DotNet Callbacks Maker
DESCRIPTION
Allow the automatic generation of Ounce 6.x rules (of type 'callback') for web services and public methods
This module parses the provided .NET dlls (directly or by recursively searching a directory) and uses .NET reflection to identify public methods or methods marked with the [WebMethod] attribute (i.e. Web Services methods)

KEY / UNIQUE FEATURES
• Quickly identify public or web services methods
• Quickly create Ounce 6.0 rules for the identified methods

USE CASES
• Scanning .NET applications with .NET Web Services

PRODUCTIZATION READINESS
CAN OSA DO THIS!

COMMERCIAL SOFTWARE DEPENDENCIES:
Connects to Ounce Rules Database

ROADMAP: NEXT DEVELOPMENT
• This module has been made redundant by the new version of Rules Manager and standalone CATNET scanning module

see presentation
for individual slides with O2 Modules details
LEGACY O2 MODULES (6X)

O2 Scanners
DESCRIPTION
O2 module that allows the easy triggering of scans using the Ounce GUI

O2 Tool - DotNet Callbacks Maker
DESCRIPTION
Allow the automatic generation of Ounce 6.x rules (of type ‘callback’) for web

O2 Tool - Findings Query
DESCRIPTION
This module provides advanced findings filtering capabilities via a LAMDA like query for example the user can use the following query to list all findings marked as vulnerabilities:

From O2Finding to O2Finding where finding.confidence == 2.0

KEY / UNIQUE FEATURES
- Ability to write filters in a dynamically constructed LAMDA query
- High performance filtering engine allows quick analysis and
  parsing of 100MB+ assessment files

USE CASES
- Filtering large LAMDA file
- Creating smaller LAMDA file based on LAMDA query results

PRODUCTIZATION READINESS
CAN O2 DO THIS?

Commercial software dependencies:
- Consumes output files created by Ounce (and others)

ROADMAP: NEXT DEVELOPMENT
- This module was made redundant by the O2 Findings Viewer Module

see presentation
for individual slides with O2 Modules details
LEGACY O2 MODULES (6X)

See presentation
for individual slides with O2 Modules details

Saturday, 5 September 2009
LEGACY O2 MODULES (6X)

O2 Scanners
DESCRIPTION
O2 module that allows the easy triggering of scans using the Ounce GUI

O2 Tool - DotNet Callbacks Maker
DESCRIPTION
Allow the automatic generation of Ounce x.x rules (of type ‘callback’) for web

O2 Tool - Findings Query
DESCRIPTION
This module provides advanced findings filtering capabilities via a LDAP file

O2 Tool - Search Assessment Run
DESCRIPTION
This was the first O2 module that fully supported the loading of multiple assessments

O2 Tool - View Assessment Run
DESCRIPTION
This was the first module to provide a simple view into the unique lists of Sources, Sinks and Links of Ounce files
This module is a simpler version of the O2 Tool - Search Assessment Run and its main use today is to provide an easier interface into the trace visualization of multiple traces

KEY / UNIQUE FEATURES
• Visualization of O2 Times
• View Finding and Trace information
• Unique set of filters for Ounce files

USE CASES
• Analysis of Ounce files

PRODUCTIZATION READINESS

CAN OS DO THIS?

Yes

Open Source (2002)

ROADMAP: NEXT DEVELOPMENT
• This module was made redundant by the O2 Findings Viewer Module

see presentation
for individual slides with O2 Modules details
LEGACY O2 MODULES (6X)

O2 Scanners
DESCRIPTION
O2 module that allows the easy triggering of scans using the Ounce GUI.

O2 Tool - DotNet Callbacks Maker
DESCRIPTION
Allow the automatic generation of Ounce 6.x rules (of type ‘callback’) for web.

O2 Tool - Findings Query
DESCRIPTION
This module provides advanced findings filtering capabilities via a LAMDA like search.

O2 Tool - Search Assessment Run
DESCRIPTION
This was the first O2 module that fully supported the loading of multiple assessment.

O2 Tool - View Assessment Run
DESCRIPTION
This was the first module to provide a single view into the unique list of

O2 Tool - WebInspect Converter
DESCRIPTION
RoC of the integration of a Black Box scanning engine (HP’s WebInspect) with a White Box scanning engine (Ounce Labs 4.5).

KEY / UNIQUE FEATURES
- Show how this integration between White Box and Black Box can be implemented.
- Create consolidated findings between multiple scan engines.

USE CASES
- Create consolidated findings between multiple scan engines.

PRODUCTIZATION READINESS
CAN OSA DO THIS?
- Yes
- With ease and blend

COMMERCIAL SOFTWARE DEPENDENCIES:
- Matches Ounce’s WebInspect’s results.

ROADMAP: NEXT DEVELOPMENT
- Add support for other Black Box scanners.
- Improve the GUI to allow the visual mapping of both set of results.
- Create command line version so that this process can be automated into a build process.

see presentation
for individual slides with O2 Modules details
**STABLE O2 MODULES (IX)**

**O2 Tool - Host Local Website**

**DESCRIPTION**
- Allows the drag and drop of a local folder which will be become exposed a locally executed web server
- This module is usually used in conjunction with the O2 Debugger Manager since once the web server has started it can be remotely hooked and instrumented.

**KEY / UNIQUE FEATURES**
- Easy creation of locally running web servers on arbitrary folders

**USE CASES**
- Debug .NET applications
- Write exploits for .NET application

**PRODUCTIZATION READINESS**

**CAN OSA DO THIS?**
- Yes
- No

**COMMERCIAL SOFTWARE DEPENDENCIES:**
- Uses Microsoft's test webpage included with Visual Studio

**ROADMAP: NEXT DEVELOPMENT**
- Add creation of ideas of webinars based on dynamic creation of unit tests

---

See presentation
VAPORWARE O2 MODULES (2X)

see presentation
for individual slides with O2 Modules details
VAPORWARE O2 MODULES (2X)

O2 Tool - Filter Assessment Files

DESCRIPTION
This module was originally written as a PoC (Proof of Concept) to deal with a number of commonly asked feature requests by Ounce users. The objective was to create a GUI that allowed the easy processing and filtering of large assessment files. (For example by extracting only the findings marked with a confidence > 99% and a severity > High). The technology required is already in O2.

KEY / UNIQUE FEATURES
• GUI-driven filtering of O2 files
• Filter O2 results in the GUI

USE CASES
• Filtering large O2 files

PRODUCTIZATION READINESS

CAN OSA DO THIS!

COMMERCIAL SOFTWARE DEPENDENCIES:
Consumes O2 files created by Ounce (and others)

ROADMAP: NEXT DEVELOPMENT
• This module is made redundant by the O2 Finding Viewer Module
• This module is currently not complete and unless there are further requests there are no extra development planned.

see presentation
for individual slides with O2 Modules details
VAPORWARE O2 MODULES (2X)

O2 Tool - Filter Assessment Files

DESCRIPTION
This module was originally written as a PoC (Proof of Concept) to deal with a number of commonly asked feature requests by Ounce users. The objective was to create a GUI that allowed the easy processing and filtering of large assessment files (for example by extracting only the findings marked with a confidence of ≤Vulnerability and a severity of High). The technology required is already in O2.

KEY / UNIQUE FEATURES
• (if implemented) Provides easy GUI-Driven filtering of Ounce files

USE CASES
• Filtering large Ounce findings files

PRODUCTIZATION READINESS
CAN OSA DO THIS?

O2 Tool - O2 Reflector

DESCRIPTION
This module aims to create a single point of decompilation/creation and visualization of .NET assemblies and Java class files. The objective is support the multiple Byte Code analysis engines and provide a single environment to quickly manually analyze compiled .NET, Java or C# files.

KEY / UNIQUE FEATURES
• Supports visualization of classes and methods for .NET assemblies (using NET Reflector and Mono C#) and for Ounce files.
• Uses MonoC# Decompiler to convert .NET Byte Code to C#.

USE CASES
• Visualize .NET assemblies, Java classes or C# files

COMMERCIAL SOFTWARE DEPENDENCIES:
Consumes Ounce files created by Ounce.

ROADMAP: NEXT DEVELOPMENT
• Add support for the latest O2 C# data creation and visualization capabilities.
• Add support for .NET code decompilation (using MonoC# decompiler or Reflector) and Java (using JAD).

see presentation
for individual slides with O2 Modules details
O2 is now an OWASP Project
“OWASP O2 PLATFORM”

OWASP Projects that O2 will immediately start to integrate with and add value:
O2 CONTRIBUTIONS TO Open Standards: ICirData, ICirClass, ICirFunction, ICir*

```
public interface ICirData
{
    Dictionary<string, ICirClass> dClasses_bySignature ( get; set; )
    Dictionary<string, ICirFunction> dFunctions_bySignature ( get; set; )
}
```

```
public interface ICirClass
{
    Dictionary<string, ICirFunction> dFunctions ( get; set; )
    Dictionary<string, ICirClass> dIsSuperClassedBy ( get; set; )
    Dictionary<string, ICirClass> dSuperClasses ( get; set; )

    string Signature ( get; set; )
    string Module ( get; set; )
    string Name ( get; set; )
    string FullName ( get; set; )
    string Namespace ( get; set; )

    // Reference to file location (or the source code in most cases)
    string File ( get; set; )
    string FileLine ( get; set; )
}
```

```
public interface ICirFunction
{
    List<ICirFunction> FunctionsCalledUniqueList ( get; set; )
    List<ICirFunctionCall> FunctionsCalled ( get; set; )
    List<ICirFunctionCall> FunctionIsCalledBy ( get; set; )
    List<ICirFunctionParameter> FunctionParameters ( get; set; )

    ICirClass ParentClass ( get; set; )
    string FunctionSignature ( get; set; )
    string ReturnType ( get; set; )
    string FunctionNameAndParameters ( get; set; )
    string ClassNameFunctionNameAndParameters ( get; set; )
    string FunctionName ( get; set; )
    string ParentClassFullName ( get; set; )
    string ParentClassName ( get; set; )
    string Module ( get; set; )
}
```

```
public interface ICirFunctionCall
{
    ICirFunction cirFunction ( get; set; )
    int lineNumber ( get; set; )
    string fileName ( get; set; )
    int sequenceNumber ( get; set; )
    String sourceCodeText ( get; set; )
}
```

```
public interface ICirFunctionParameter
{
    String ParameterName ( get; set; )
    string ParameterType ( get; set; )
    string Constant ( get; set; )
    bool HasConstant ( get; set; )
    bool HasDefaultValue ( get; set; )
    string Method ( get; set; )
    bool IsTainted ( get; set; )
}
```
TWO O2 WEBSITES

• Like OWASP’s SAMM (& others), in the short term, O2 will be hosted in two separate websites:

  • **Official** 'stable' versions will be hosted using OWASP's WIKI engine at: http://www.owasp.org/index.php/OWASP O2 Platform

  • **Development** versions & Community features will be hosted using SquareSpace web engine at: http://www.o2-ounceopen.com
BTW, SOMEBODY should sponsor an OWASP ‘Application Security Summit’ :)”

- Which would be a world wide gathering of security experts with the objective to figure out how to use the current resources (People, Process and Technology) to help customers to fix security vulnerabilities in their applications

- This Summit could be organized by OWASP using the same model used on the last OWASP Summit in Portugal
THE CHALLENGE
• For this discussion a ‘Framework’ is an environment which augments the capabilities of the core language implementations (.NET Framework or J2EE). Examples of what I call a Frameworks are: Spring, Struts, Microsoft Enterprise Library, SharePoint, WebSphere Portal, SalesForce API,

• Each Framework creates its own ‘reality’ almost like a VM (Virtual Machine), where they (for example Spring MVC) create an abstraction layer between the core language (i.e. Java) and the target application.

• So, if the scanning engines (Black Box, White Box, Human Brain) don’t explicitly support frameworks, they will NOT understand how they work they and will NOT be able to find security issues in the applications built on top of those frameworks.
  • It is like trying to use a C++/Binary analyzer to scan JITTED .NET code (i.e. the assembly representation of .NET code)
SOME TECHNOLOGICAL SOLUTIONS THAT STILL NEED TO BE SOLVED

• All current (Commercial and Open Source) Static Source Code Analysis tools have most (if not all) of the problems below (some have minor/basic coverage of it)

• ANALYSIS ENGINEs - Part I
  • Attributes, Collections & other type of objects that receive taint in A and output it in B
  • Global Variables
  • Proper Taint Propagation across strings and between data types
  • Reflection (which creates ‘Hyper Jumps’ between code paths)
  • Events
  • Rules based on assemblies/jars versions and not on signatures
  • Taint Typing (also applied to business logic)

• ANALYSIS ENGINEs - Part II
  • Rules Management (user-friendly process to mass create, edit, modify, import and export)
  • Join Traces (between application layers or interfaces or ‘Hyper Jumps’)
  • Read (and understand) configuration files (who have major impact on the attack surface and exploitability)
  • Auto Attack Surface Markup
  • Expose Control Flow
  • Understand Framework behavior

• GlassBox
  • Integration with WB & BB (driving one tool from the other)
  • Common Reporting

• Note: this (list above)
  IS A VERY SMALL & LIMITED LIST of the technologies / techniques that need to be supported when running (manual or automatic, Black or White) scans.
  These capabilities (either when used by non-expert users or by expert security consultants) allows the security engagement to be accurate, effective, consumable and actionable
WHERE WE ARE TODAY
and WHERE WE NEED TO BE ASAP

• Here is the evolution of technologies and where the current level of support is:

  • 1996-2000: MainFrames, Web Servers, Java, ASP Classic
  • 2000-2004: C/C++, .NET Framework, J2EE, PHP
  • 2004-2006: Struts, Spring Framework, Ajax, Flash, Hibernate, Microsoft Enterprise Library
  • 2006-2009: lots of web innovation going on, here is a small list:

    **Languages & Technologies:** Aspect, Web Services, REST, Widgets/Gadgets, AIR, Silverlight, Groovy & Grails, Python, Ruby & Ruby on Rails, JSP EL, Velocity, JSF (Faces),

    **Application Platforms / Frameworks:** ASP.NET MVC, SharePoint, IBM WebSphere Portal, WebSphere Application Portal, SAP (web stuff), iPhone & Apple iStore

    **Online Applications:** SalesForce, Amazon Web Services, MySpace/FaceBook/Twitter

    **OWASP ‘standards/APIs/frameworks’:** ESAPI, SAMM, ASVF, etc...

    And let’s not forget that most enterprise applications have their OWN frameworks and APIs (and sometimes even VMs)

    • 2010-....: Chrome, cloud computing (vSphere (VMWare’s cloud), Azure (Microsoft’s cloud)), Web 3.0 and next generation of all of the above :)

  ‘Out of the box’ capabilities is here

  O2 is here

  We need to be here ASAP

Saturday, 5 September 2009
TO SCALE WE NEED TARGETED SOLUTIONS
HOW TO SCALE: AUTOMATE SECURITY KNOWLEDGE

• The only way we will be able to scale (and have these solutions used by a wide audience (from developer’s upwards), is if we are able to ‘capture + automate’ the knowledge, workflow and wisdom of security consultants. And we need to do this in such a way that repeated analysis by non-technical staff will have the same result has the analysis created by an security expert.

• In a nutshell ... what we need is to do,

    is to automate the security expert’s brain ...

    so that we are able to independently use it in a repeatable and consistently way,

    and once we have done that (automating their brain) ... we can work on making it

    very simple to use by non-security experts

    And due to the complexity of each targeted application / framework ...

    ... this ‘one button’ solution is only possible if ....

WE CREATE TARGETED SOLUTIONS & PRODUCT

(see next 4 slides for an example of what this could look like)

Note that today an ‘Application Security Analysis’ engagement is a very: complex, non-repeatable, non-scalable, non-measurable, and very opaque (from the client point of view) process. It is also very hard to calculate its ROI.
Due to the complexity and ‘realities’ created by the Spring Framework, the only way to deal to analyze/expose its behavior is to create fine-tune ‘packages’ of the available technology.
SHAREPOINT (MOSS) : SECURITY ANALYSIS PLATFORM

- Same think for frameworks & development environments like Microsoft Office Sharepoint Server (MOSS). Unless we have a customized engine & technology that understands Sharepoint, it is very hard (if not impossible) to (for example) write secure web parts.
• .... and the same thing applies for applications built on top MOSS (which also create their own reality and unique class of vulnerabilities (before & after customization)

• quote from www.shareworkz.com: "... ShareWorkz helps you get the most from Microsoft SharePoint – quickly! Built in SharePoint Server 2007 Standard Edition, ShareWorkz reduces the time to build and deploy a best practice, enterprise class SharePoint 2007 Solution to 1 month or less..."
• The Open Source community also needs a generic platform made up of only Open Source or free tools.
• This is a very CRITICAL piece of the puzzle, since this is what will enable the wide use of these techniques across the Open Source and Commercial Software development world (it will also allow the Framework developers to be responsible for creating their markups (after all, who better than the Spring developers to help with the development of the “Spring Framework : Security Analysis Platform”)

Saturday, 5 September 2009
Thank you ....