Introduce SCADA vulnerability and a little suggestion for vulnerability numbering format

Kai – Chi Chang (K.C.)
2012/11/12
Botnet@icst.org.tw
NESOKING@Gmail.com
Outline

- My Organization
- The Basic
  - CVE & OSVDB
  - SCADA
- Our Experience
- My propose
- Conclusion
My Organization

- I work for III (Institute for Information Industry)
- My department is Cyber Trust Technology Institute
  - Information Security Service Center
CVE & OSVDB

- **CVE (Common Vulnerabilities and Exposures)**
  - That is the most popular vulnerability database
  - It is good for searching, but the CVE number includes few information

- **OSVDB**
  - That is open source vulnerability database, it focuses on open source software
  - The goal of the project is to provide accurate, detailed, current and unbiased technical information on security vulnerabilities.
  - It is good for searching, but the OSVDB number includes few information
Easy Webinar Plugin for WordPress contains a flaw that may allow an attacker to carry out an SQL Injection attack. The issue is due to the `get_widget.php` script not properly sanitizing user-supplied input to the 'wid' parameter. This may allow an attacker to inject or manipulate SQL queries in the back-end database, allowing for the manipulation or disclosure of arbitrary data.

Location: Remote / Network Access
Attack Type: Input Manipulation
Impact: Loss of Integrity
Solution: Solution Unknown
Exploit: Exploit Unknown
Disclosure: Third-party Verified, Uncoordinated Disclosure
OSVDB: Web Related

OSVDB is not aware of a solution for this vulnerability.

References:
- Exploit Database: 22300

Manual Testing Notes:

```
http://[target]/wp-content/plugins/webinar_plugin/get-widget.php?wid=[SQL]
```
Sometimes, It doesn’t work

Hacker's command

Victim's Report

XML-RPC Interface

Introduction

This document contains information on the XML-RPC interface that Nucleus provides, and the error messages it spits out. Please note that the specification of this interface might still undergo changes in the future.

The URL for the Nucleus XML-RPC interface is:
What is SCADA?
(Supervisory Control and Data Acquisition)
- It is a part of ICS (Industrial Control System)
- If it could monitor system and extract the data then it is SCADA

In the different domain, it could have different functions. It should have the basic unit
- Human Machine Interface, HMI
- Monitor System and Extract Data
- Remote Terminal Unit, RTU
- Programmable Logic Controller, PLC
- Communication infrastructure
How does SCADA work?

- SCADA could Control hardware and extract data by Computer, PLC and User Interface
  - In powerhouse, SCADA needs to collect a lot of information.
  - Those information include voltage, temperature, humidity Etc.
  - To monitor those information and record those data.
  - According to the information, SACADA could regulate voltage in the real time

Why do we need that?

Only in the powerhouse?

Where can find SCADA?
- Stuxnet & Duqu
  - Stuxne (2010) and Duqu Worm (2011)
    - All of them, the attack target is SCADA
    - Hacker infect USB device or malicious file spread other

Ref. http://www.youtube.com/watch?v=LwCU0B6PMi0&feature=share
Our experience for revealing vulnerability

- Notify the CERT about 36
  - Notify ICS-CERT: 26 zero day
  - Notify JPCERT/CC: 9 zero day
  - Notify CERT/CC: 1 zero day

- We get the 36 CVE identifier number
Vendor

- SIEMENS
- Invensys
- EMERSON
- GE Intelligent Platforms
- Schneider Electric
- Citect (Schneider Electric)
- 7-Technologies (Schneider Electric)
- ARC Informatique
- Beijer Electronics
- Mitsubishi Electric
- Advantech
- ADLINK
- ...

[Logos of various vendors]
Our experience for revealing vulnerability

- We can be found some acknowledgment on the ICS-CERT website
Our experience for revealing vulnerability

<table>
<thead>
<tr>
<th>Disc. Date</th>
<th>OSVDB ID</th>
<th>CVEID</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-09-28</td>
<td>83522</td>
<td>2012-3035</td>
<td>Deltav Malformed String Parsing Remote Overflow DoS</td>
</tr>
<tr>
<td>2012-05-16</td>
<td>82014</td>
<td>2012-1918</td>
<td>Deltav Multiple Product Unspecified ActiveX Arbitrary File Overwrite</td>
</tr>
<tr>
<td>2012-05-16</td>
<td>82011</td>
<td>2012-1915</td>
<td>Deltav Multiple Product Unspecified SQL Injection</td>
</tr>
<tr>
<td>2012-05-16</td>
<td>82012</td>
<td>2012-1916</td>
<td>Deltav Multiple Product PORTSERV.exe Packet Parsing Remote DoS</td>
</tr>
<tr>
<td>2012-05-16</td>
<td>82013</td>
<td>2012-1917</td>
<td>Deltav Multiple Product Project File Handling Remote Overflow</td>
</tr>
<tr>
<td>2012-05-16</td>
<td>82019</td>
<td>2012-1914</td>
<td>Deltav Multiple Product Unspecified XSS</td>
</tr>
<tr>
<td>2012-02-17</td>
<td>79402</td>
<td>2012-0223</td>
<td>7-Technologies TERMS Unspecified Path Subversion Arbitrary DLL Injection Code Execution</td>
</tr>
<tr>
<td>2012-02-17</td>
<td>79405</td>
<td>2012-0224</td>
<td>7-Technologies TERMS Unspecified Path Subversion Arbitrary DLL Injection Code Execution</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79553</td>
<td>2012-0234</td>
<td>Advantech/Broadwin WebAccess Unspecified SQL Injection (2012-0234)</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79570</td>
<td>2012-0235</td>
<td>Advantech/Broadwin WebAccess Unspecified CSRF</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79532</td>
<td>2011-4521</td>
<td>Advantech/Broadwin WebAccess Unspecified SQL Injection (2011-4521)</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79556</td>
<td>2012-0233</td>
<td>Advantech/Broadwin WebAccess Unspecified XSS</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79559</td>
<td>2012-0236</td>
<td>Advantech/Broadwin WebAccess Unspecified Information Disclosure</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79574</td>
<td>2012-0237</td>
<td>Advantech/Broadwin WebAccess Unauthorized Date/Time Syncing Modification</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79575</td>
<td>2012-0238</td>
<td>Advantech/Broadwin WebAccess opctrl . asp Remote Overflow</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79577</td>
<td>2012-0239</td>
<td>Advantech/Broadwin WebAccess uaddupadmin . asp Unauthorized Admin Password Manipulation</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79579</td>
<td>2011-4524</td>
<td>Advantech/Broadwin WebAccess Unspecified Overflow</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79582</td>
<td>2011-4526</td>
<td>Advantech/Broadwin WebAccess Unspecified ActiveX Overflow</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79587</td>
<td>2011-4525</td>
<td>Advantech/Broadwin WebAccess Arbitrary File Write Remote Code Execution</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79567</td>
<td>2011-4522</td>
<td>Advantech/Broadwin WebAccess bwerrdn . asp Unspecified XSS</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79568</td>
<td>2011-4523</td>
<td>Advantech/Broadwin WebAccess bwview . asp Unspecified XSS</td>
</tr>
<tr>
<td>2012-02-16</td>
<td>79595</td>
<td>2012-0244</td>
<td>Advantech/Broadwin WebAccess Unspecified SQL Injection (2012-0244)</td>
</tr>
<tr>
<td>2012-01-16</td>
<td>73323</td>
<td>2011-4053</td>
<td>7-Technologies Interactive Graphical SCADA System Path Subversion Arbitrary DLL Injection Code Execution</td>
</tr>
<tr>
<td>2011-12-21</td>
<td>73233</td>
<td>2012-0309</td>
<td>Cogent DataHub Unspecified XSS</td>
</tr>
</tbody>
</table>
That is CVE number

That is the CVE number


May I have an easier way to understand what it means?
My propose

- The idea of vulnerability numbering format
  - It is possible to have a easy way to know the meaning of the number
    ◎ If it is possible, It should have the vulnerability’s OS, the attack type of POC and time
  - If it is possible, base on the CVE and OSVDB is better
    ◎ If we do that, all of vulnerability could be integrated

- Any Cert in FiRST can reveal vulnerabilities
  - We may have ability to find the vulnerabilities, and we want to notify other organizations (It takes a lot of time)
The Idea of vulnerability numbering format

<table>
<thead>
<tr>
<th>New</th>
<th>3</th>
<th>0</th>
<th>0</th>
<th>6</th>
<th>1</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSVDB</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>1</td>
<td>8</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>OSVDB-85544</td>
</tr>
<tr>
<td>CVE</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>CVE-2012-3011</td>
</tr>
</tbody>
</table>
The Idea of vulnerability numbering format
The Idea of vulnerability numbering format
The Idea of vulnerability numbering format

**New**

```
1 2 1 1 0 0 0 0 0 0 1
```

**OSVDB**

```
1 2 0 9 1 8 5 5 4 4
```

**CVE**

```
1 2 0 9 2 0 3 0 1 1
```

- **Year**
- **Month**
- **Identify Flag**
- **Sequence Number of Vulnerabilities**
Any Cert in first can reveal vulnerabilities.
Any Cert in first can reveal vulnerabilities
Conclusion

● The SCADA issue is more and more important.
  – We will still use fuzzing technology to find more vulnerabilities in the future.
● CVE and OSVDB are most popular vulnerability database, but the number is difficult to identify some information.
● Using the new format, it could identify
  – Which kind of OS
  – Which attack type of POC
  – And, it base on CVE and OSVDB number
● At last, if any Cert reveal vulnerabilities to FiRST, It would shorten the time of notification
Thank you for your kind attention
Q&A