Host based anomaly detection for webservers

RP1

Sudesh Jethoe

Overview

1. Introduction
2. Problem description
3. Research Questions & Method
4. Analyze
5. Solutions
6. Result
7. Conclusion

Introduction

ComputerWeekly.com

News | IT Management | Industry Sectors | Technology Topics | Blogs | Multimedia | Vendor Content | Jobs

Home > Topics > IT security > Hackers and cybercrime prevention > HSBC back online after DDoS attack

NEWS

HSBC back online after DDoS attack

Warwick Ashford
Friday 19 October 2012 11:31

Share +1 0 Tweet 16

HSBC has restored its online banking services after a distributed denial of service (DDoS) attack.

HSBC said servers had come under a DDoS attack which affected HSBC websites around the world.

The DDoS attack on HSBC did not affect any customer data, but did prevent customers using HSBC online services, including internet banking.

"We are cooperating with the relevant authorities and will cooperate with other organisations that have been similarly affected by such criminal acts," HSBC said.

Byte Internet

- Since 1999
- Managed hosting
  - Shared hosting
- 10.000+ sites
  -

Overview

1. Introduction
2. Problem description
3. Research Questions & Method
4. Analyze
5. Solutions
6. Result
7. Conclusion

Problem description

Facts:

- Sites get hacked
- Sites get abused
  - spam
  - malware distribution
  - (d)dos

Cause?

old versions of:

- frameworks
- plugins

weak passwords

What can customers do

- Update web application frameworks
  - Joomla, Wordpress
- Avoid buggy plugins
  - guestbook, photoalbum
- Use encrypted channels for data-transport

  ssh vs ftp

Why customers do not:

Dependency on customers

- Unaware
- Don't know how
- Don't want to risk it
- Unable/unwilling to pay for security measures

Overview

1. Introduction
2. Problem description
3. Research Questions & Method
4. Analyze
5. Solutions
6. Result
7. Conclusion

Research Questions

Can we develop a method which detects interactive malware (for example a webshell) running on servers in a shared hosting environment?

- What are the characteristics of this kind of malware?
- How can the characteristics be used to detect this malware?
- How do existing solutions detect this malware?
- Can we make use of existing frameworks for the detection and prevention in a hosting providers environment?

Method

- Collect malware
- Run it in a controlled environment
- Collect logs
- Review existing solutions
- Integrate method in a suitable solution

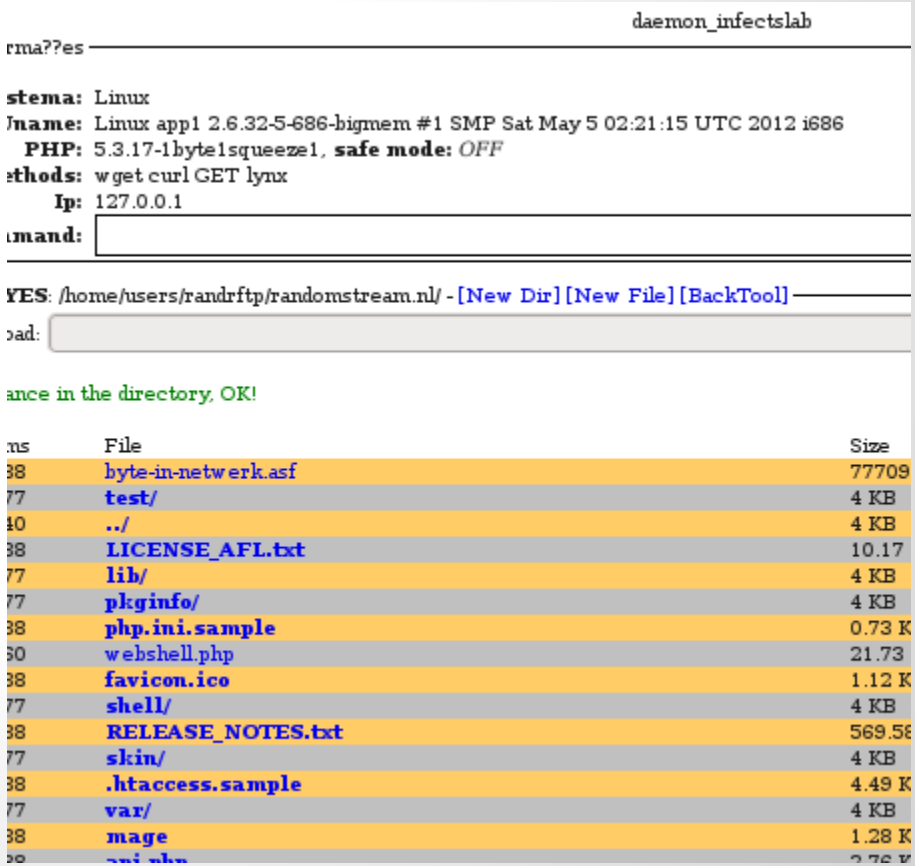
Overview

1. Introduction
2. Problem description
3. Research Questions & Method
4. Analyze
5. Solutions
6. Result
7. Conclusion

Cases (1/3) johanstegels.nl & webcast.nl

<form method=\"**POST**\" action=\"{$fstring}&action=save&chdir={$chdir}&file={$file}\">

randomstream.nl 188.142.*.* - - [25/Oct/2012:11:37:11 +0200] "**POST** /webshell.php?http://www.education.zp.ua/images/down.jpg?&action=cmd&chdir=/home/users/randrftp/randomstream.nl/ HTTP/1.1" 200 3835 "http://randomstream.nl/webshell.php?http://www.education.zp.ua/images/down.jpg?&action=cmd&chdir=/home/users/randrftp/randomstream.nl/" "Mozilla/5.0 (X11; Linux x86_64; rv:16.0) Gecko/20100101 Firefox/16.0"



Cases (2/3) florian.nl

indx.php

```
switch($_POST['action'])
{
	case "upload":UploadFile($_FILES['file']);
	break;
	case "stop":stoped();
	break;
**snip**
}
```

46.21.*.* web10.c4 www.florian.nl - - [18/Oct/2012:14:34:19 +0200] "**POST** /shop//langs/nl/indx.php HTTP/1.1" 200 - "-" "-" "-" "-" 46.21.145.228 florian.nl pid:31699 1608779 0 0 32002 36002

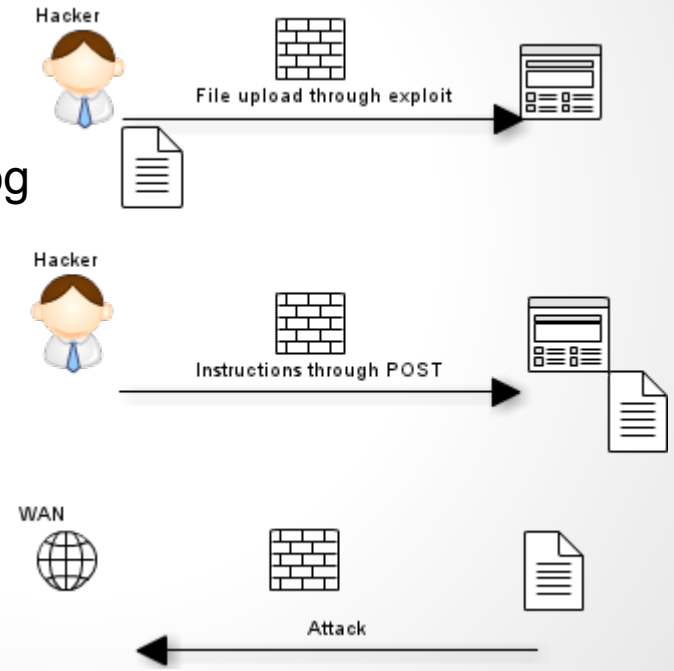
Cases (3/3) liverunning.nl

Cases (3/3) liverunning.nl

199.15.*.* web8.c2 liverunning.nl - - [18/Oct/2012:12:05:39 +0200] "**POST /index.php**?option=**com_phocaguestbook**&view=phocaguestbook&id=2&Itemid=248 HTTP/1.0" 200 25805 "Mozilla/5.0 (Windows NT 5.1; rv:9.0.1) Gecko/20100101 Firefox/9.0.1" "-" "-" 199.15.*.* liverunning.nl

Analyze

1. Hacker abuses exploit
2. Hacker uploads malicious script
3. Hacker instructs script
    a. POST is used
        i. no character limit
        ii. content not shown in log
4. Malicious script is executed



Detect?

POST analysis
7 sites, 7 days

Site	urls POSTed to	real files POSTed to
sc******	451	13
it******	37	0
fa******	198	12
de******	0	0
dm*****	410	0
aa*****	344	1
aa******	130	2

Overview

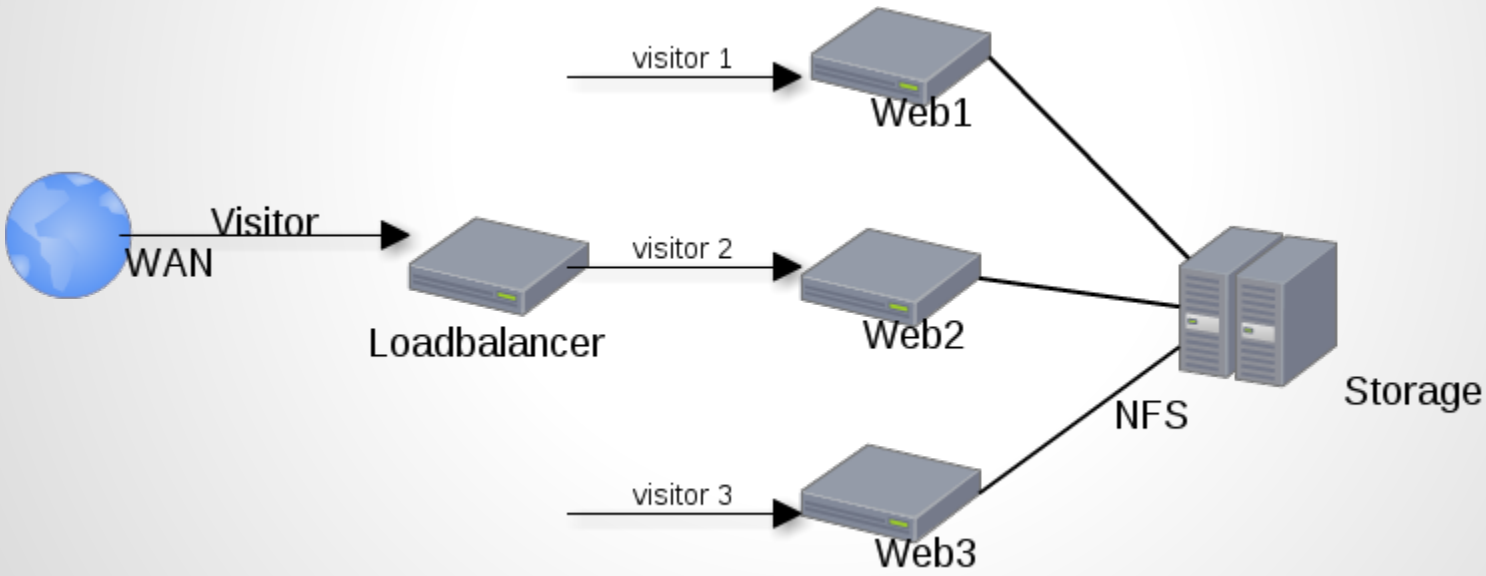
1. Introduction
2. Problem description
3. Research Questions & Method
4. Analyze
5. Solutions
6. Result
7. Conclusion

Solutions (Hosting Provider)

- Network Intrusion Detection Systems (NIDS)
- Web Application Firewalls (WAF)

- Host Intrusion Detection Systems

Byte Internals



Solutions (Hosting Provider)

Network Intrusion Detection System

+ Can detect (and block) uploads in early stages
- Does not work on encrypted channels
- Depends on signatures (only detects known malware)

Web Application Firewalls

+ Can be finetuned to look for specific instructions
- Inspection takes time and slows visitor experience

Host Intrusion Detection Systems

+ Integrated tools for checking various system variables (files,logs)
- Not suitable for working over a LAN

Overview

1. Introduction
2. Problem description
3. Research Questions & Method
4. Analyze
5. Solutions
6. Result
7. Conclusion

Result

byte-security-POST-IDS

1. generate whitelist of files which can be posted to
2. tail access.log
3. grep POST
4. test files for:
    a. included in whitelist
        i. modifications
5. alert

Overview

1. Introduction
2. Problem description
3. Research Questions & Method
4. Analyze
5. Solutions
6. Result
7. Conclusion

Conclusion

- malicious scripts can be detected
- not suitable for attacks on indirect urls

Future work

- Tweak whitelist flagging
  - Who maintains the whitelist?
    - Site maintainers
    - The hosting provider
    - An algorithm?
- Read rewrite rules to find more files
  - For example by enabling mod_rewrite logging in Apache