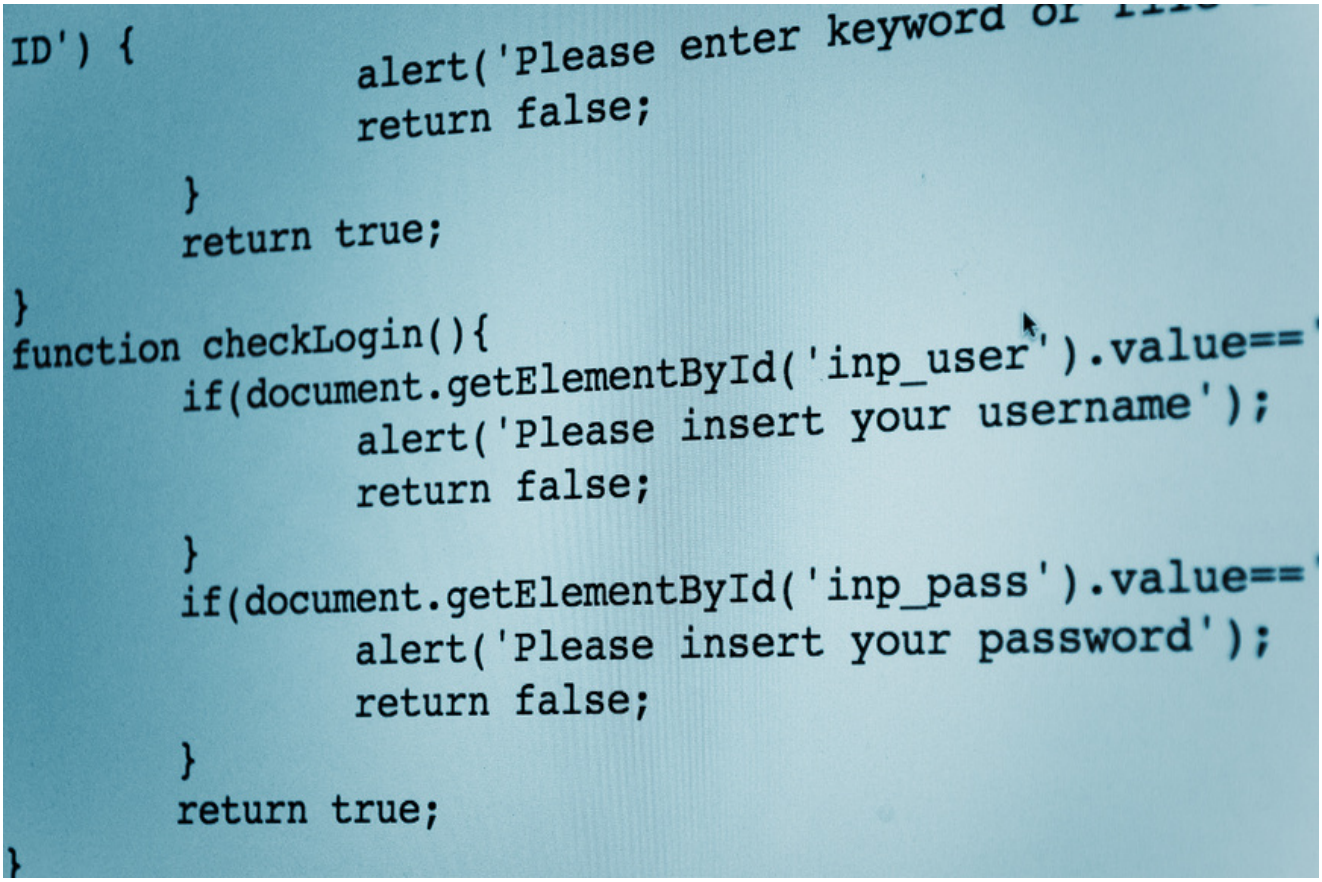
Win32/Spy.Ranbyus modifying Java code in RBS Ukraine systems

welivesecurity.com/2012/12/19/win32spy-ranbyus-modifying-java-code-in-rbs/

December 19, 2012



Win32/Spy.Ranbyus shows how it is possible to bypass payment transaction signing/authentication with smartcard devices and has started to modify java code in one of the most popular remote banking systems (RBS) in the Ukraine.

19 Dec 2012 - 09:37AM

Win32/Spy.Ranbyus shows how it is possible to bypass payment transaction signing/authentication with smartcard devices and has started to modify java code in one of the most popular remote banking systems (RBS) in the Ukraine.

I've already mentioned the Win32/Spy.Ranbyus family in my previous blog post about smartcard monitoring in modern banking malware (Smartcard vulnerabilities in modern banking malware). It displays really interesting functionality because it shows how it is possible to bypass payment transaction signing/authentication with smartcard devices. We have been tracking the latest modification to this malware family and the trojan Ranbyus

has started to modify java code in one of the most popular remote banking systems (RBS) in the Ukraine, BIFIT's *iBank 2*. ESET Virus Radar statistics show that Ukraine is the region most affected ever by Ranbyus infection.

This banking trojan doesn't have web-injection functionality and instead implements a targeted attack on specific banking/payment software. Win32/Spy.Ranbyus collects information about the infected system (active processes, OS version and so on) and forwards it to its command center. The main functionality for stealing money is based on a set of various form grabbers for specific payment software. For example, grabbers for software developed for the java platform look like this:

```
int __usercall inject_javaw<eax>(int a1<ebx>)
{
  int v1; // eax@3
  char v3; // [sp+0h] [bp-10h]@1

  init(&v3);
  create_process_mutex(&v3);
  if ( get_config_value_96() )
    get_import(&v3);
  delete_iBank_files();
  check_BIFIT_keys(a1);
  mem_alloc();
  init_java_hooks(&v3);
  init_javaw_grabber();
  init_java_javaw_browser_grabber();
  v1 = get_imports_table();
  (*(v1 + kernel32_Sleep))(0xFFFFFFFFu);
  exit_process(&v3);
  return 0;
}
```

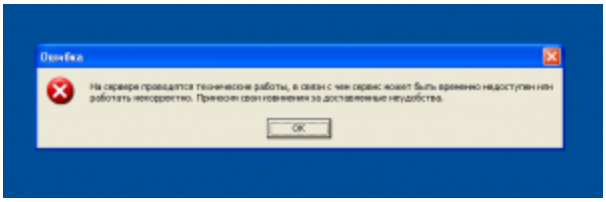
```
int __cdecl inject_java()
{
  int v0; // eax@3
  char v2; // [sp+0h] [bp-10h]@1

  init(&v2);
  create_process_mutex(&v2);
  if ( get_config_value_96() )
    get_import(&v2);
  mem_alloc();
  delete_iBank_files();
  init_java_hooks(&v2);
  init_browser_and_java_grabber();
  init_java_javaw_browser_grabber();
  create_thread_ex(sub_4074B9, 0, 0);
  v0 = get_imports_table();
  (*(v0 + kernel32_Sleep))(-1);
  exit_process(&v2);
  return 0;
}
```

I've already disclosed information about java patching functionality in another banking malware family, Carberp (Carberp Gang Evolution: CARO 2012 presentation). Carberp has specific functionality for modifying the JVM (Java Virtual Machine) and tracking payment software activity. And Ranbyus is based on a different method, modifying java code only for specific application without changing the JVM. For example, Ranbyus can modify the balance figures so as to hide information about fake transactions implemented through the malware.

```
removeNodeFromParent•
(Ljavax/swing/tree/MutableTreeNode;)V•
getText•
getUserObject•
getChild•
(Ljava/lang/Object;I)Ljava/lang/Object;•
getChildCount•
(Ljava/lang/Object;)I•
getRoot•
getModel•
()Ljavax/swing/tree/TreeModel;•
class com.bifit.swing.tree.XTree•
getComponent•
(I)Ljava/awt/Component;•
()Ljava/lang/Object;•
getComponentCount•
()I•
javax/swing/text/JTextComponent•
javax/swing/JLabel•
prepareRenderer•
(Ljavax/swing/table/TableCellRenderer;II)Ljava/awt/Component;•
getCellRenderer•
(II)Ljavax/swing/table/TableCellRenderer;•
getContentPane•
()Ljava/awt/Container;•
SUM•
(I)Ljava/lang/String;•
%.02f•
repaint•
setText•
(Ljava/lang/String;)V•
class javax.swing.JLabel•
fireTableDataChanged•
(I)Ljava/lang/Object;•
getRowCount•
getColumnCount•
()Ljavax/swing/table/TableModel;•
class com.bifit.swing.table.Table•
```

[Tracked java methods used by Win32/Spy.Ranbyus]

In addition, Win32/Spy.Ranbyus can block RBS software activity and show the following message in the Russian language:



Translated from the Russian the message looks like this:

"*Technical work is being performed on the server, and the service may be temporarily unavailable. We apologize for the inconvenience*".

Ranbyus targets Ukrainian and Russian banks and is never seen in campaigns targeting other regions. The command center panel for the Win32/Spy.Ranbyus botnet looks like this:

The Carberp gang is the (crime) market leader in the Russian region and has already secured a safe position in the top 20 most active threats in Russia for a full year (Carberp, the renaissance). Ranbyus has the leading position among banking malware in the Ukrainian region.

The SHA1 hash for the Win32/Spy.Ranbyus.I dropper mentioned here is:
ee6c14f26962447a30823f9f8d20a53d29322617

Special thanks to my colleagues Anton Cherepanov and Dmitry Volkov (Group-IB).

Aleksandr Matrosov, Security Intelligence Team Lead

19 Dec 2012 - 09:37AM

Sign up to receive an email update whenever a new article is published in our Ukraine Crisis – Digital Security Resource Center

Newsletter

Discussion