

The Hushmail Report

"A. Smith", December 2007

Hushmail has been the web's leading provider of encrypted web mail for many years, I started using Hushmail in early 2001, at that time Hushmail did not have any known flaws, and the safer Java login was the only alternative (which was good!), they were the best, with the best features, like open-source, however this has now changed due to the facts that users have reported that some accounts have been compromised.

Note: if you want to know more about the background, you can read it here:
<http://blog.wired.com/27bstroke6/2007/11/encrypted-e-mai.html>
http://www.theregister.co.uk/2007/11/08/hushmail_court_orders

In October 2007 some of my friends started to ask me questions like: "-Is Hushmail still safe?", as a security consultant, I wanted to investigate this further...

First I checked the registration routine:

When you register a new account there are two different registration types, I call them type 1 and type 2:

Account type 1: Non-java as default

This is the new Default registration, probably since early 2006. This type is not as safe as type 2 because the passphrase leaves your computer unprocessed, as plaintext (inside SSL).

If you registered your account with type 1 you should be aware that your passphrase **could** be logged. In this account-type all encryption/decryption are performed by Hushmail's server, you put all trust in the hands of Hushmail. This registration type is also more vulnerable to MITM attacks (packet sniffing).

See picture 1.

Note: The HTTPS protocol (HTTP with SSL/TLS) is only protecting the communication between you and the server, if someone have access to the server they can store your password, bank-account or whatever you are sending.


Account type 2: Java-enabled

This type is almost hidden on the registration page, you have to click on [Show advanced options] to see it. After that you must click on [Enable Java] and [Gather random Data] to enable java and the best security level. This account-type is the safest since the passphrase is hashed inside the Java applet before it leaves your computer. In this account-type all encryption/decryption are performed directly on your computer.

See picture 2.

Note: more about the hashing routine can be found in RFC 4880, 3.7.1.3

Picture 1: Account type 1, non-Java (lower security)

Address  <https://www.hushmail.com/hushmail/index.php>



New Secure Email Account

Welcome to Hushmail, the world's premier free, secure web-based email and document storage system.

Step 1

Choose your new email address: @hushmail.com

[Click here to use an automatically generated email address](#)

Step 2

The security of your account is determined by the strength of your passphrase. Please use a passphrase that is much longer than an ordinary password. For advice on generating a strong passphrase, see <http://www.diceware.com>.

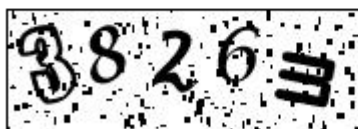
Choose your passphrase:

Re-type your passphrase:

Step 3

Five numbers are displayed below to help us distinguish between real people like you and computer programs trying to use our service.

Please type the five numbers you see below:




[Click here if you cannot read these numbers](#)


Step 4 (Optional)

[Show advanced options](#)

Step 5

Picture 2: Account type 2, Java-enabled (higher security)

Address  <https://www.hushmail.com/hushmail/index.php>



New Secure Email Account

Welcome to Hushmail, the world's premier free, secure web-based email and document storage system.

Step 1

Choose your new email address: @hushmail.com

[Click here to use an automatically generated email address](#)

Step 2

The security of your account is determined by the strength of your passphrase. Please use a passphrase that is much longer than an ordinary password. For advice on generating a strong passphrase, see <http://www.diceware.com>.

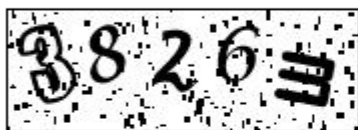
Choose your passphrase:

Re-type your passphrase:

Step 3

Five numbers are displayed below to help us distinguish between real people like you and computer programs trying to use our service.

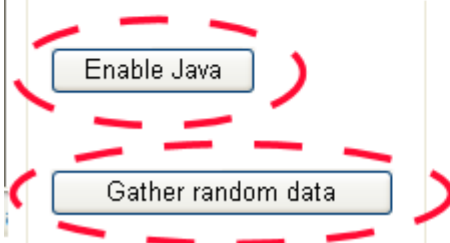
Please type the five numbers you see below:



[Click here if you cannot read these numbers](#)

Step 4 (Optional)

[Hide advanced options](#)



The Java version of the Hush Encryption Engine provides end-to-end encryption using a Java applet, however many computers are not compatible with this technology.

If Java is enabled, random data can be collected based on the movement of your mouse to further protect your security.

After you have finished your registration you can't go back and change the account-type, it's only Hushmail who can change your account-type, it does not matter if you clear your cookies.

When you login to your account, you can have different login screens, if you have the safer type-2 account you will only have one login screen, see picture 3. If you registered your account without Java (type-1 account) you can have two different login screens, see picture 4 and 5.

Java login with **account-type 2** is the best alternative (picture 3).

If you have **account-type 1** and non-Java login screen (picture 4) you can click on [Enable Java] to use the safer login, you will then see picture 5, which is good. But even if you have done this, your account-type will not be as safe as account-type 2, because your account registration process was different, and you can't change the past. Another weakness with account-type 1 is that the Java on/off setting is stored in a cookie, if you clear your cookies you will get back to non-Java login screen (picture 4).

How do I know that there is no back door in the Java-applet?

If it existed, even the Java-login would be unsafe, but so far I have not found any back door in the applet, I have spent several hours on it, using a SSL-MITM-attack to read the packets that the applet is sending, and I also spent a few hours reading the source code, as far I can see, the standard applet, with SHA-1 hash: b55a5308989636e26c037a1bb8fefe7714832173 (version 2.5.0.9) do not contain any back door.

Ok, but how did they decrypt the accounts that were compromised?

This is my theory:

1. Hushmail get a court order from the Supreme Court of British Columbia to target specific account.

Note: Remember that several countries all over the world have a mutual legal assistance treaty with Canada, if some government in your country convinced the Supreme Court of British Columbia, they will send a court order to Hushmail.

2. Hushmail will start logging your login procedure to get the passphrase.

If you have **account-type 2** they will **switch your account to type-1**, so if you are sure that you registered an type-2 account and suddenly have a different login screen, as picture 4, **you know that Hushmail have changed your account.**

If you have **account-type 1** with Java enabled (pic 5) they will probably remove your cookie settings to get you back to the unsafe non-Java login (pic 4).

If you forget to check the java settings on the login screen, then you're dead meat! IF Hushmail is targeting your account...

Three different logins screens

Picture 3. Java login - Account type 2 (safest)



Please enter your passphrase:

Authenticate

Picture 4. non-Java login - Account type 1 (unsafe!)

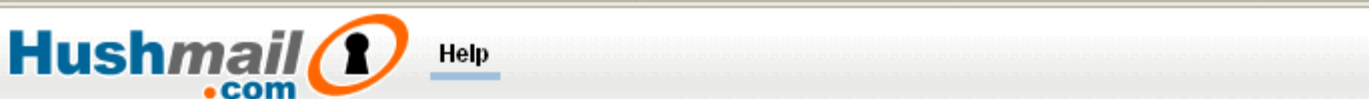


Please enter your passphrase:

Authenticate

[Enable Java](#) | [Explain](#)

Picture 5. Java login - Account type 1 (safe login, unsafe account)



Please enter your passphrase:

Authenticate

[Turn off Java](#) | [Explain](#)

I have checked many different Hushmail accounts, 7 of them belonged to people who was busted by the police, **all** of these 7 accounts was set to non-Java login!, and **account-type 1**.

Ok, but what if they send you a "fake" Java-applet, with a back-door?

So far I have not seen any fake applet at all, but you can verify that by doing two steps:

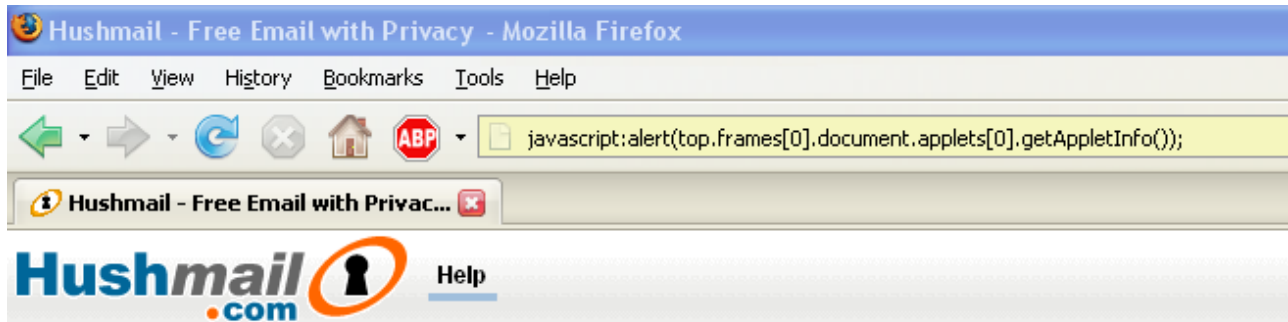
Step 1.

On your login screen (pic 3),

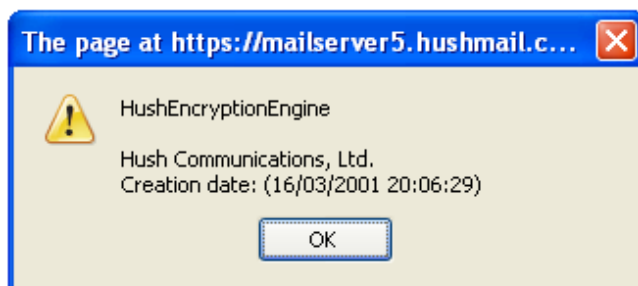
copy and paste this in your URL field:

[javascript:alert\(top.frames\[0\].document.applets\[0\].getAppletInfo\(\)\);](javascript:alert(top.frames[0].document.applets[0].getAppletInfo());)

and press Enter, you should now see a popup window:



Please enter your passphrase:



If you don't see any popup window, first check that the javascript string was pasted correctly, if you still don't get any popup window, something is wrong, **do not login**.

If you can see the popup, click Ok to close the window and go to step 2, but If you're paranoid, you should also make sure that the function `authenticateUser()` is more than 3 lines long by viewing the Page Source, I recommend using FireFox to do this, right-click on the passphrase textfield and select: This Frame > View Frame source, press CTRL-F and enter `authenticateUser`, you should now see the `function authenticateUser()`, scroll down, if the function looks like below, **do not login**.

```
function authenticateUser()
{
    return true;
}
```

This function should be much longer than 3 lines, if the beginning of the function looks like below, it should be ok!

