Template:User committed identity

From Wikipedia, the free encyclopedia

Committed identity: **{{{1}}}** is a SHA-512 commitment to this user's real-life identity.

Template documentation

This template gives you a way to later prove that you are the person who was in control of your account on the day this template was placed. This is done by putting a code (called a "hash") on your user page so that, in the event that your account is compromised, you can convince someone else that you are really the person behind your username.

Contents

Why?

The intended use of this template is to help in the hopefully unlikely event that your account is compromised. If you published your real-life identity, then that identity could be used to reestablish contact with you if your account were compromised; keep in mind, in this scenario contact could not be established with you through your account, since it may be under the control of someone else. However, many Wikipedia users do not disclose their real-life identities, or disclose little enough of them that it may be difficult to establish their identity.

This is not a replacement for having a strong password, nor for registering an email address for your account. You should still do everything you can to prevent your account being compromised, including using a strong password and remembering to log yourself out when using a computer to which others may have access. If you have one, it may also be helpful to post your PGP public key. But even with the best of precautions, your account could become compromised, for instance, via a trojan horse or a brute-force attack on your password. This is intended to be a last resort.

How

The idea is to use cryptographic hashes; you choose a secret string known only to yourself, put it through a one-way hash function, and publish the result somewhere. It is infeasible to determine the secret string corresponding to the hash; hence, an attacker compromising an account presumably would not be able to supply the secret string.

Syntax

```
{{User committed identity|hash|hash function used|background=CSS color|border=CSS
color|article=grammatical article for the hash function}}
```

Italicized text should be replaced with appropriate input, or its parameter should be removed. Parameters are represented by "parameter=*value*", and separated by vertical bars |.

- Replace "hash" with the hash produced from your secret string. This unnamed parameter is equivalent to a parameter named "1" (see parameters).
- The "hash function used" parameter, if not included, defaults to SHA-512. (This hash function is strongly recommended.)
- The "background" parameter, if not included, defaults to #E0E8FF (light blue, see Web colors#Hex triplet)
- The "border" parameter, if not included, also defaults to #E0E8FF.
- The "article" parameter, if not included, defaults to "a". The other likely value is "an".

For example, if your hash is "ef7c4c55a176bd20ed558aaefde21c4803080195" using SHA-1, and you want a light orange box with a black border, use the following code:

```
{{User committed identity|ef7c4c55a176bd20ed558aaefde21c4803080195|SHA-
1|background=#FC9|border=#000}}
```

to produce

Committed identity: **ef7c4c55a176bd20ed558aaefde21c4803080195** is a SHA-1 commitment to this user's real-life identity.

Choosing a good secret string

1. Your secret string should end with a long string of random text, such as "fFfwq0DuDmMXj8hYTM3NTKeDhk". This ensures that brute force and dictionary attacks cannot infer your identity from your public hash.
2. Your secret string should specify enough of your identity that, if the string were revealed, you could unambiguously prove you match that identity. At least two means of contact is a good rule. For instance, your secret string could include a telephone number and email address at which you can be reached. However, it should *not* contain data that you are not willing to show to Wikipedia's administrative staff.
3. Try not to choose a secret string that represents your identity that could go completely out of date. For instance, it may be bad to choose a string that specifies *only* your telephone number as that number might change.
4. If you want to change your secret string, do so, but keep track of all your old secret strings. It is best to reveal all of them if you ever want to confirm your identity, as this will establish that you are the same person who used your account from the first moment the committed identity was published.
5. Advanced options:
   - If you have public accounts on other websites with different passwords, list URLs of those accounts. You can later take a specified action to prove that you own those accounts. For example, if you have a YouTube account, an administrator can provide a string which you then insert in a video comment.
   - You may include information such as your driver's license number, national identification number, or passport number. You can then later supply copies of these documents as additional evidence to prove your identity.
   - Another option is to take a photo or video of yourself, take a SHA hash of the resulting file, and include that hash in your secret string. Retain the file. You can then later supply the file to an

administrator, and they can video call with you and compare the file with your current appearance. This will remain effective even if the attacker has compromised all your listed means of contact.

Example

Bare minimum: at least two forms of contact and a random string:

```
joe@example.com 555-123-3456 fFfwq0DuDmMXj8hYTM3NTKeDhk
```

More complete example: full name, multiple forms of contact, contact information for trusted friends, and a random string:

```
Joe Schmoe. joe@example.com. 555-123-3456. P.O. Box 1234, San Jose, CA. My best friend Bob's email: bob@example.c
```

After Joe uses the secret string, he can generate a new secret string with a new hash by merely changing the random characters at the end.

A comprehensive example including multiple trusted people and advanced options:

```
Joe Schmoe. joe@example.com. 555-123-3456. P.O. Box 1234, San Jose, CA.
My best friend Bob's contact info: bob@example.com, 555-234-5678
My wife's contact info: anne@example.com, 555-345-6789
Other accounts: http://www.youtube.com/user/joeschmoe http://flickr.com/photos/joeschmoe/
Driver's license number: 123456789, SSN: 123-45-6789, passport number: 9876H432L
SHA-512 of joeschmoe.jpg: 747ec1836486a3dbe8a5d6805a2cc080fb8dc427d9535579ecb04c750d7a4a515641fd0411ed6bb97242a3e
NSffKWSHaGbcTm3WGtE8hyUQ
```

In this case, Joe would supply both the above string (e.g. as a text file) and joeschmoe.jpg to administrative staff. Staff might ask him to post a specific comment on YouTube, to send a scan of his passport, or to video call to confirm that his appearance matches joeschmoe.jpg.

Getting the hash

Be sure to note the *exact string* you enter into the form, in case you need to use it later. It is important that this string be both easily remembered exactly by you and hard to guess or find by any intruder - if an intruder knows the secret string, then this scheme is useless and provides no security. One's username is public and trivially guessable; one's password is not a good choice either, as in the event of a compromised account, the password is likely to have been guessed.

There are various methods you can use to obtain the hash of your secret string. Some methods use websites, while others use only software running on your computer (local). The pros and cons of both are described below.

Security

Web-based methods require you to enter your secret string on a web page that could potentially be collecting that information. However, if your secret string includes information about things that you control, as recommended, someone who knew your secret string and your Wikipedia username would still be unable to take control of your Wikipedia account. Thus, the security risk associated with web-based methods is rather low. Also consider that any personal details in your secret string could be revealed to operators of the website. If these factors are unacceptable to you, use a local method.

Ease of use

Local methods are more difficult to use and may require you to install additional software on your computer. Web-based methods are largely self-explanatory: you simply copy and paste your secret string into a box on a web page and click a button or link. The website calculates the hash and displays it, and you copy and paste it into your `{{User committed identity}}` template. You need only do both copy-and-paste operations correctly, being careful not to drop or add any characters.

To verify any SHA-512 hashing method, use it to generate the hash for the following string:
`My name is Joe Schmoe, and I can be contacted at: joe@example.com`
The SHA-512 hash of that string should be as follows:

```
b7a84efbbd843545666957384e874c894fdc17f48ced53abd231c2e4d08e45ad10287b1225432e3ed9794c12994ff1e82aecf66a2ded61ad4baf6d8b9c81dab8
```

Web-based methods

One of the websites you can use to easily generate a SHA-512 hash is this one (http://hash.online-convert.com/sha512-generator). Copy and paste your secret string into the first box, then click the "Convert file" button. The site will display a box containing your hash in four formats: hex, HEX, h:e:x, and base64. Use the first format, labelled "hex:", and ignore the remaining three formats. Copy and paste the 128-character hash value, not including the "hex:" label. The box is horizontally scrollable, which requires you to select the value left-to-right, causing the box to scroll. When the entire value is selected (highlighted), copy it using Ctrl+C, ⌘ Cmd+C, or other method appropriate for your type of computer. Then, paste the value into your template.

Local methods

On Unix-like operating systems the `sha1sum`, `sha224sum`, `sha256sum`, `sha384sum`, and `sha512sum` programs are provided in the GNU Core Utilities.

Windows users may use one of the methods mentioned below; those who have PowerShell installed can generate a hash using the following command. (The template defaults to SHA-512 if the hash function parameter is omitted.) Provide your secret phrase in the location indicated:

```
[bitconverter]::tostring((new-object
security.cryptography.sha512managed).computehash([text.encoding]::utf8.getbytes("Secret
phrase here"))).replace("-", "")
```

Users of Windows 7 and Windows 8.1, in which PowerShell is installed by default, can generate a hash using the Get-FileHash command. A 512-bit hash can be generated by saving your secret phrase in a text file, indicated here as the file "secret.txt", located in the "C:\Users\JoeShmoe" folder, and entering the following command:

```
PS C:\Users\JoeShmoe> Get-FileHash secret.txt -Algorithm SHA512 | Format-List
```

It is recommended that SHA-512 be used, as recent cryptographic research has cast doubt on the long-term security of SHA-1. For security, you should only use locally executed programs, or client-side javascript, to create your hash. Examples of such hash calculators include SHA512 Encrypt (http://coursesweb.net/javascript/sha512-encrypt-hash_cs), jsSHA (http://caligatio.github.com/jsSHA/) and HashCalc 2.01 (http://www.slavasoft.com/hashcalc/).

Compromised accounts

In case your account is compromised, to make use of your committed identity to someone and prove you are the same person who originally controlled the account, give a trusted user the *exact secret string* you originally entered into the box. They can then compute the appropriate hash of that same string and verify that it is the same result and that you are who you say you are.

Once you've established your identity, and set up a new account or regained control of the original account, you'll probably want to create a new hash as now someone (possibly multiple someones depending on how and to whom you communicated the secret string) else knows the secret string.

Weakness

This technique of establishing identity by revealing the secret string behind the hash is not inherently attack proof; it increases attacker effort substantially (if the secret string is chosen and handled properly) which is worthwhile, and does so at little effort to the legitimate user. But it is attackable in that anyone can invent their own secret string, hash it, and claim an identity.

An attacker with access to the account could overwrite the hash with their own one. They could then say that the previous owner of the account was an attacker who claimed their identity and generated his own hash.

An attacker without access to the account could claim that the current account's owner stole their identity. The attacker could state that they did not publish a hash when they used to own the account, or that they did not register an account and that someone else is using their name.

This weakness does not indicate that the commitment scheme is worthless, because the *commit phase* did not apply to all interested parties (the real person and all potential attackers).

See also

- Don't leave your fly open (essay)
- User:Unimaginative Username/Simple Committed ID Instructions
- Strong password
- Wikipedia:Changing username
- Wikipedia:Security
- Wikipedia Signpost: Committed identity

Code	Result	Transclusions
{{User:Anomie/Userbox committed identity|…}}	This user has an SHA-512 **committed identity**. See this userbox's invocation (https://en.wikipedia.org/w/index.php?title=Template:User_committed_identity&action=edit).	Transclusions (https://en.wikipedia.org/w/index.php?title=Special:Whatlinkshere/User:Anomie/Userbox_committed_identity&limit=500)
{{User:Urdna/CIDuserbox}}	This user's account is secured with **a unique Committed Identity.**	Usage (https://en.wikipedia.org/w/index.php?title=Special:WhatLinksHere/User:Urdna/CIDuserbox&hidelinks=1&hideredirs=1)
{{Template:User CID}}	This user account is secured with a **unique committed identity**.	Usage (https://en.wikipedia.org/w/index.php?title=Special:WhatLinksHere/Template:User_CID&hidelinks=1&hideredirs=1)

The above documentation is transcluded from Template:User committed identity/doc. (edit (https://en.wikipedia.org/w/index.php?title=Template:User_committed_identity/doc&action=edit) | history (https://en.wikipedia.org/w/index.php?title=Template:User_committed_identity/doc&action=history))
Editors can experiment in this template's sandbox (edit (https://en.wikipedia.org/w/index.php?title=Template:User_committed_identity/sandbox&action=edit) | diff (https://en.wikipedia.org

/w/index.php?title=Special%3AComparePages&page1=Template%3AUser+committed+identity&page2=Template%3AUser+committed+identity%2Fsandbox)) *and testcases* (create (https://en.wikipedia.org/w/index.php?title=Template:User_committed_identity/testcases&action=edit&preload=Template%3ADocumentation%2Fpreload-testcases)) *pages.*
Please add categories to the /doc subpage. Subpages of this template.

Retrieved from "https://en.wikipedia.org/w/index.php?title=Template:User_committed_identity&oldid=388764494"

Categories: User namespace templates

- This page was last modified on 4 October 2010, at 17:16.
-