
Table of Contents

Introduction	1.1
What is Phishing?	1.2
Password-Stealing Phishing	1.2.1
OAuth Phishing	1.2.2
How to spot Phishing	1.3
How to prevent Phishing	1.4
Two-Factor Authentication	1.4.1
Security Keys	1.4.2
Review your Accounts	1.5
Conclusions	1.6
Reading Material	1.7
License and Credits	1.8

Guide to Phishing

While phishing is a form of attack that has been around for a long time, functional and usable solutions have been pretty hard to identify. Why is it hard to eradicate? Because of its simplicity. Because we have grown accustomed to the process of authenticating with passwords to online services that are recognizable by their brand, it can be hard to be alert of the possibility that a legitimate-looking login form that we have naturally filled hundreds of times before could be malicious.

Because of its simplicity, popularity, and effectiveness, phishing has naturally become a primary tactic against human rights defenders, dissidents, and journalists all over the world. Unfortunately, security education programs often fall behind the current trends, and attackers are fast to adapt and defeat widely spread recommendation.

This guide is intended for individuals at risk and security trainers who wish to learn more in depth the modern strategies and tactics used in phishing attacks, and available mitigations.

Note: this guide is currently under development. You can contribute to this text [here](#).

What is Phishing

Note: Across articles and security guides the terminology might vary, in fact some people refer to "phishing" as attacks that are untargeted in nature, and "spear-phishing" for attacks that are instead targeted, regardless of whether it is credentials phishing or a malware attack. For the purpose of this guide, we will more simply use "phishing" to refer to credentials phishing.

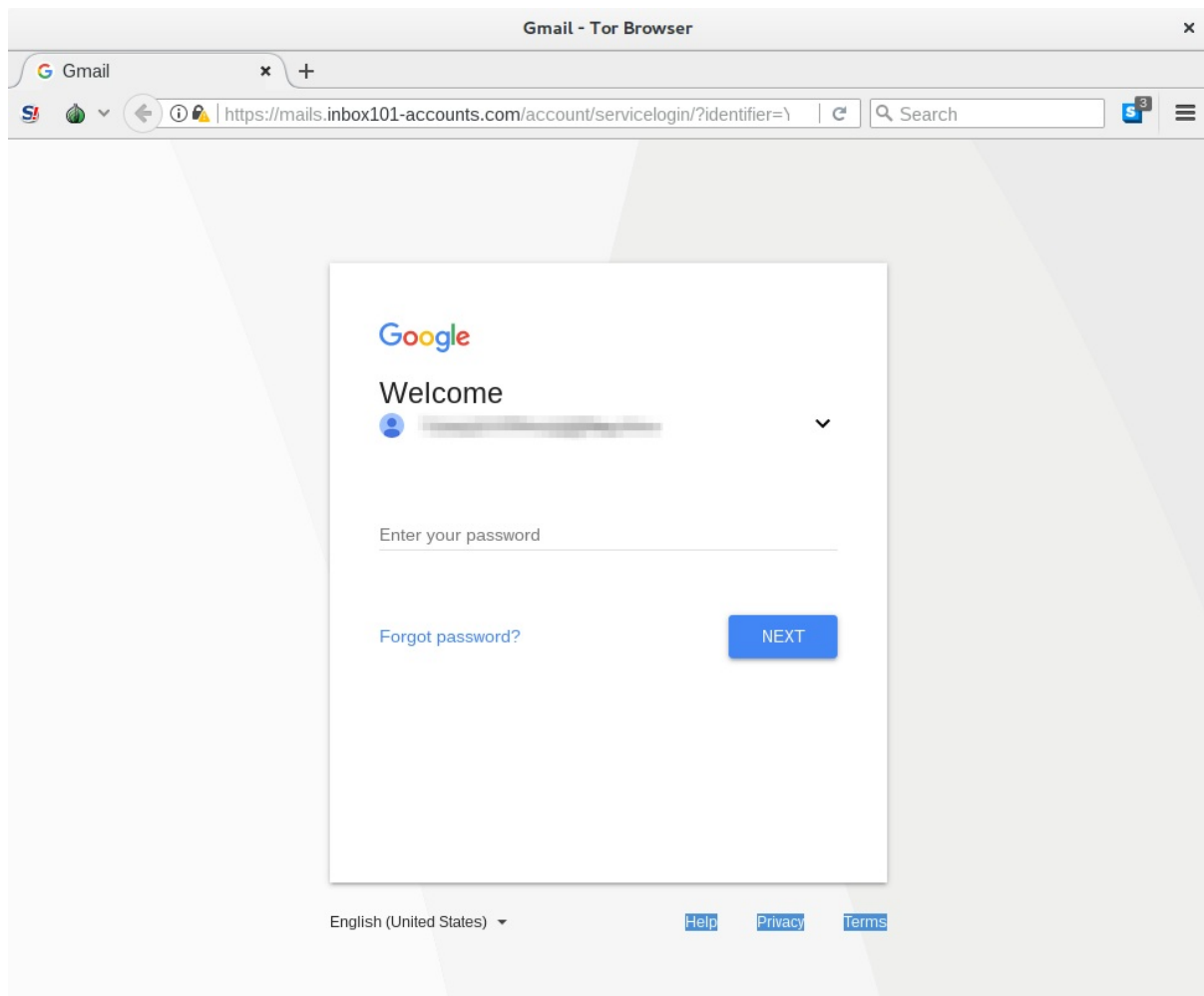
Phishing is a form of digital attack with the objective of obtaining access to a target's email, social media or other online accounts. The reasons why attackers recur to phishing can be various, and the primary certainly is because especially email accounts are particularly appetible sources of information for an attacker. Additionally, obtaining access to emails or other online accounts could allow to, for example, impersonate the victim in order to conduct further attacks or even obtain access to yet more accounts through a password recovery process.

There are different types of phishing attacks, but essentially we can categorize them in two:

- Password-Stealing Phishing
- OAuth Phishing (or Third-Party Application Phishing)

Password-Stealing Phishing

Password-Stealing Phishing is the most traditional form of phishing attack. It relies on tricking the target into providing their passwords by luring them to clones of the login prompts of a legitimate site. These fakes are normally generated by modifying HTML templates that resemble as much as possible the original cloned site. Following is an example of a rather realistic-looking clone of Gmail used in a targeted phishing campaign in the Middle-East:



Screenshot by [Amnesty International](#)

The quality and sophistication of this form of phishing attack depends on the attacker's attention to detail in the creation of the clones and on the extent to which the phishing kit emulates the behavior of the original website. Consequently, this classic form of phishing attack can range from being trivially obvious to very deceitful. For example, the more dedicated attackers would create phishing pages pre-compiled with the target's email address and even the target's profile picture, as well as any other detail to lower any suspicion. Even better attackers, as we discuss more in detail later, might also be capable of bypassing the most common forms of two-factor authentication.

Although it doesn't properly fit into this category, an evolved version of this form of attack is **session riding** or **session hijacking**. With session riding the attacker, instead of having to recreate as accurately as possible a clone of the original site, creates a "reverse proxy" that simply sits in between the target and the legitimate service, and is able to intercept session tokens that allow them to authenticate to the victim's account. This technique is for example implemented in the open source tool Evilginx2 and is explained in greater detail [here](#).

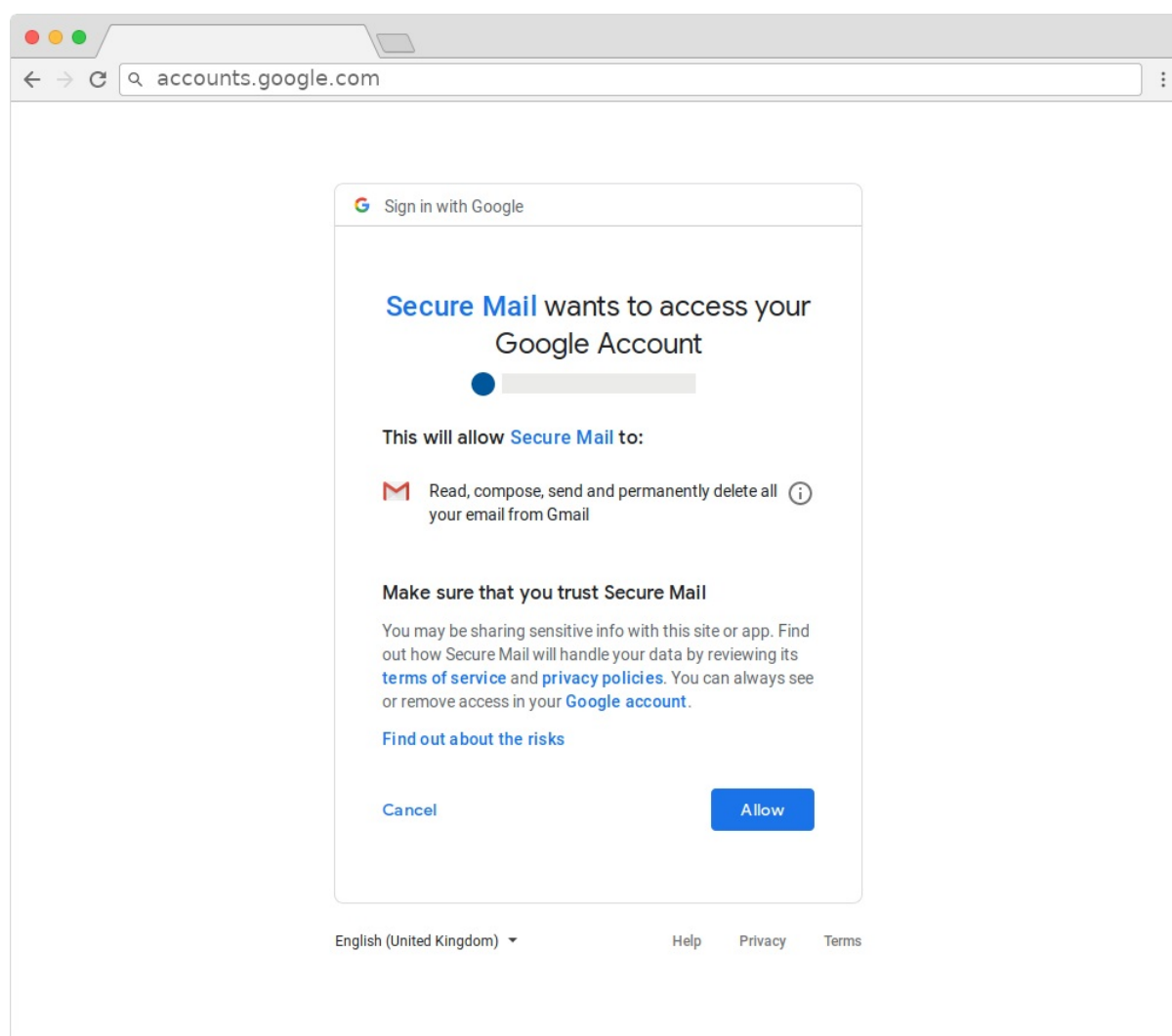
Because session riding relies on intercepting authentication to the original service website (such as Gmail or Facebook), the *only* visible clue that the website they are visiting is not the original is the domain name in the browser's address bar.

OAuth Phishing (or Third-Party Application Phishing)

OAuth Phishing is an insidious form of phishing that does not rely on luring a target to a fake clone of a legitimate site, but rather on abusing an existing feature most online services provide. Google, Yahoo and others, for example, allow their users to connect their account to a third-party app that can be granted privileges to read the content of their email inboxes and more. For example, a legitimate third-party app could leverage this to allow users to import flight bookings or other reservations from their email inbox to a separate calendar application. You can read more about this functionality [here](#).

Attackers have been abusing this feature by creating malicious third-party apps that are masqueraded as legitimate (and generally directly offered by the service provider) and tricking the targets into allowing the application access to their accounts. While with this form of phishing the attackers do not actually obtain the password to the account, the effect remains the same, because by getting granted the right privileges, they are able to read the emails of the victims anyway through the third-party app.

Following is a screenshot of a malicious third-party app called "Secure Mail" requesting access to a victim's Gmail account.



Screenshot by [Amnesty International](#)

While normally users are warned by Google and others when they are about to grant access to a third-party application, OAuth Phishing is a much lesser known tactic and can be quite deceitful. Additionally, because the authentication to the account happens through the legitimate service, traditional anti-phishing mitigations (such as those we will explain later in this text) do not provide any advantage here.

How to spot phishing?

Because of the fact that technological mitigations have been lagging behind, the security and privacy community have over the years devised a combination of tips that should have enabled individuals at risk to identify phishing attacks. Whether it is to look at the address bar of the browser to identify the correct domain name, or to look for the green padlock on the left of it, these recommendations are hard to follow systematically and often become obsolete (and rather dangerous) as phishing attacks evolve and as browsers evolve too.

For example, at this point, the infamous green padlock to the left of the address bar in browsers (which indicates the availability of SSL/TLS, or more simply of the ability to navigate a website over the more secure `https://` instead of `http://`) is not a useful indicator of a legitimate website anymore. In the last years, services like [Cloudflare](#) and [Let's Encrypt](#) made the adoption of SSL/TLS a lot more accessible to everyone, unfortunately including attackers launching phishing attacks (which is an inevitable downside of services that are otherwise greatly beneficial to the public.)

Similarly, checking for the right domain name in the address bar (for example, checking if "google.com" is explicitly visible) might be misleading.

Both these shortcomings are well demonstrated in this example:

Quick phishing demo. Would you fall for something like this? pic.twitter.com/phONMKHBlE

— Mustafa Al-Bassam (@musalbas) [September 9, 2018](#)

As the video shows, while at first sight the webpage appears legitimate (there's the good old green padlock, a mark for "Secure" and the domain in the address bar appears to be "accounts.google.com"), expanding the window shows that the actual domain name was way longer ("accounts.google.com.secure.computer.shop") but hiding behind the limited view size. This trick is extremely common in real phishing attacks, and it works very well especially when targeting mobile users, because mobile browsers are necessarily limited in size and it is not trivial to inspect the expanded domain name.

Ultimately, if you know where to look and what to expect, phishing attacks can generally be identified through various visual indicators, but they can be more deceitful than you'd think! It might be useful to train your eye in recognizing anomalies, and [this quiz](#) from Google might help.

Two-factor authentication

An attempted mitigation to phishing has been developed over the years and many (although likely not most) have by now enabled what is commonly referred to as **two-factor authentication**, **two-factor verification**, **two-step verification**, etc. The basic idea is quite simple: because passwords can be stolen or guessed, you will be required to perform a secondary verification that should prove your ownership of the account for which a login is being attempted.

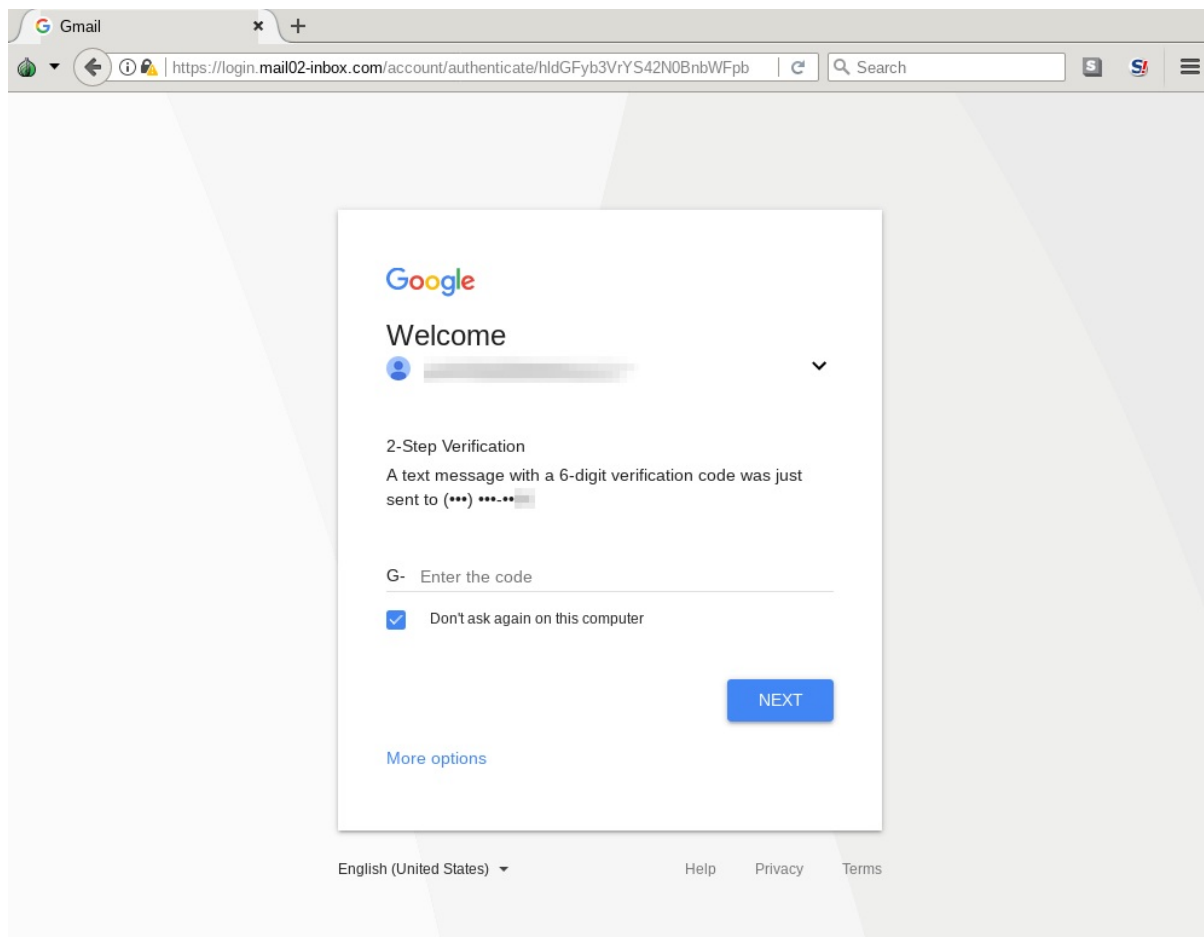
Different websites offer implement different flavors of two-factor authentication, but most commonly they come in the following forms:

1. **SMS verification:** After having successfully entered your credentials, the service (such as Google, Yahoo, Facebook, or Twitter) will send an SMS message to the phone number you provided upon registration, containing a token (generally numerical or alphanumeric) which you will be required as well to enter in the login page.
2. **Authenticator apps:** When you enable two-factor authentication, some services will ask you to install a mobile app on your phone which you will be required to open when attempting a login to fetch a short-lived numerical or alphanumeric token to enter in the login page, very similarly to the previous option.
3. **Push notification:** When you enable two-factor authentication, some services will ask you to install a mobile app which will automatically send you a notification when a login to your account is attempted, and if you wish to authorize it you'd just need to tap a button.

Without any doubt, these forms of two-factor verification are helpful mitigations that could thwart casual attackers from being able to illegitimately access your accounts. These procedures can be particularly effective against non-targeted phishing attacks (more simply, the kind of attack that would be sent to people en masse through spam emails) and against password reuse (which could be a very real risk considering the [many data breaches](#) we keep learning about.)

However, **these forms of two-factor authentication are not a solution to all phishing**. If an attacker is sufficiently resourceful (and many are) they can automate their phishing platform to effectively bypass two-factor verification. Quite simply, if you have fallen for such an attack and have already provided your username and password, the attackers can simply solicit a valid token from you just in the same way. They would then automatically use the token before it expires, and successfully log into your account.

For example, following is a screenshot of a phishing site soliciting a verification code that was sent via SMS:



This is not hypothetical. We have seen several cases of large-scale targeted phishing campaigns that have employed this technique. For example in the following reports:

- [When Best Practice Isn't Good Enough: Large Campaigns of Phishing Attacks in Middle East and North Africa Target Privacy-Conscious Users](#) by Amnesty International
- [The Return of The Charming Kitten](#) by CERTFA.

Please note: this is not to say that those forms of two-factor authentication are useless. Not at all. [If a service you are using provides any of those options to you](#), make sure to enable it as it is nevertheless a useful additional layer of security. However, if you are an individual at risk, who might especially be targeted by persistent attackers, you might want to explore additional options (such as those explained later) and generally exercise a lot more caution.

Security keys

Currently, security keys are the most secure form of two-factor authentication. Security keys are hardware tokens that implement what is known as the [U2F protocol](#) and they generally come in the shape of USB keys, although some support other channels such as NFC (mostly to function with mobile devices). When an online account is protected using a security key, the user is required not only to enter their password but to physically insert the security key in the device. This process is nicely visualized in the following animation taken from the [Solo](#) Kickstarter campaign:

While in the case, for example, of SMS verification an attacker could steal the verification code sent to the user, in this case the attacker can only physically steal the security key (as well as obtain the password) in order to log into the target's account. Obviously, this is significantly harder.

More considerations on two-factor authentication and security keys are available in [this post](#) from Amnesty International:

This technology is supported for example by Google's Advanced Protection program, by Facebook and as of recently by Twitter as well. This process might appear painful at first, but it significantly raises the difficulty for any attacker to be successful, and it isn't quite as burdensome as one might think. Normally, you will be required to use a security key only when you are authenticating for the first time from a new device.

That said, security keys have downsides as well. Firstly, they are still at a very early stage of adoption: only few services support them and most email clients (such as Thunderbird) are still in the process of developing an integration. Secondly, you can of course lose your security key and be locked out of your accounts. However, you could just in the same way lose the phone you use for other forms of two-factor authentication, and in both cases, you should carefully configure an option for recovery (through printed codes or a secondary key) as instructed by the particular service.

There are numerous companies producing these security keys. When you shop for one, make sure that your selected option explicitly supports U2F. Some of the options we recommend are:

- [YubiKey](#)
- [NitroKey](#)
- [SoloKeys](#) (open hardware and open source)

Review your Accounts

Occasionally it is a good practice to review the privacy & security settings or any suspicious activity on your accounts.


On Google you can visit the [Privacy Checkup](#) and [Security Checkup](#) pages. These pages would normally highlight any misconfigurations or any suspicious events.

For example, it might highlight malicious third-party applications!

← Apps with access to your account


Third-party apps with account access

You gave these sites and apps access to some of your Google account data, including info that may be sensitive. Remove access for those that you no longer trust or use. [Find out about the risks](#)

**Secure Mail**
Has access to Gmail

REMOVE ACCESS

Has access to:

 Gmail
Read, compose, send and permanently delete all your email from Gmail

ⓘ Basic account info

View your email address
Know who you are on Google

Homepage: ⓘ

https://mail-secure.online

Access given to: ⓘ

user-members.pro

Access given on:

1 minute ago

See something suspicious? [Report this app](#)

Screenshot by [Amnesty International](#)

You might also find devices connected to your accounts that you do not recognize, or that you have not been using in a long time. A more comprehensive list of security settings for your Google account to review can be found [here](#). We recommend you give it a thorough read.

Facebook offers a similar security checkup at [this page](#).

Conclusions

Ultimately there is no perfect solution to phishing, because the techniques used by attackers vary so much and because they become more creative by the day. There are a number of available mitigations that can effectively help, but that might not always be available to you or that might not always fit your particular use case.

Always be cautious when you are solicited a login into one of your accounts,

Reading Material

Following are some investigative reports describing various phishing campaigns targeting civil society around the world:

- March 29, 2012: [Syrian Activists Targeted With Facebook Phishing Attack](#) - EFF
- February 2, 2017: [Nile Phish: Large-Scale Phishing Campaign Targeting Egyptian Civil Society](#) - Citizen Lab
- February 14, 2017: [Operation Kingphish: Uncovering a Campaign of Cyber Attacks against Civil Society in Qatar and Nepal](#) - Amnesty International
- May 25, 2017: [Tainted Leaks: Disinformation and Phishing With a Russian Nexus](#) - Citizen Lab
- January 30, 2018: [Spying on a Budget](#) - Citizen Lab
- May 15, 2018: [Human Rights Under Surveillance - Digital Threats Against Human Rights Defenders in Pakistan](#) - Amnesty International
- December 19, 2018: [When Best Practice Isn't Good Enough: Large Campaigns of Phishing Attacks in Middle East and North Africa Target Privacy-Conscious Users](#) - Amnesty International
- March 6, 2019: [Phishing attacks using third-party applications against Egyptian civil society organizations](#) - Amnesty International

License and Credits

This guide was written by [Claudio Guarnieri](#) between 2018 and 2019. This guide is licensed as [Creative Commons Attribution-ShareAlike 4.0 International License](#).