**SPAM, SCAMS, AND OTHER FALSEHOODS**

**Introduction**

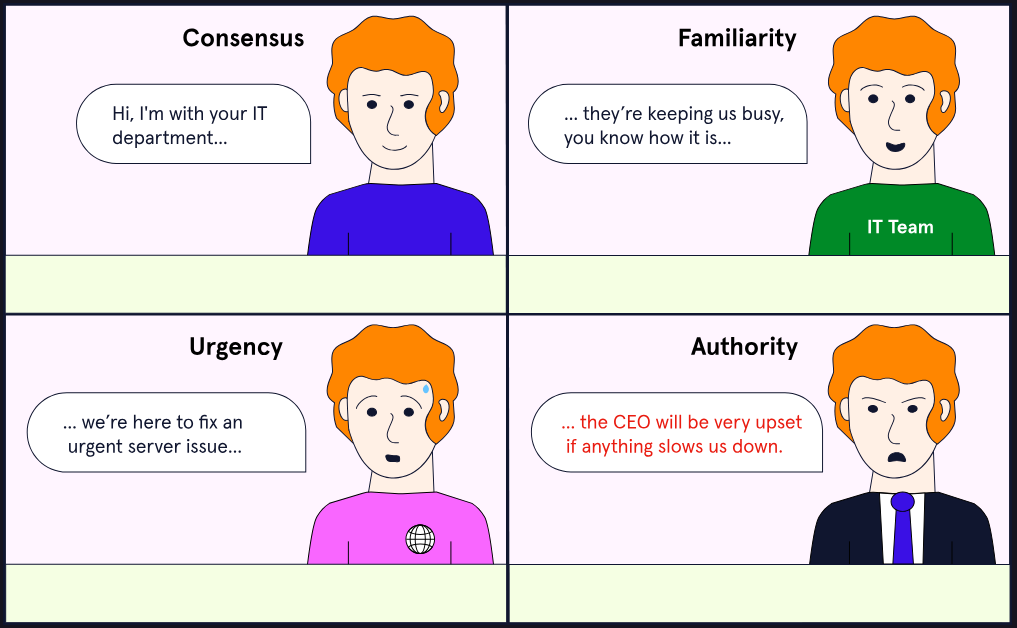
Social engineering tactics usually utilize the following principles:

* **Consensus**: when a social engineer convinces victims that they have already been trusted by others.
* **Familiarity**: when a social engineer uses charisma and likability to get a victim to complete a request.
* **Urgency**: when a social engineer creates a sense of urgency or scarcity to put time pressure on a victim.
* **Authority**: a high-risk strategy in which a social engineer attempts to intimidate a victim or claim authority over them.

While these principles describe the high-level concepts of [social engineering](https://www.codecademy.com/resources/docs/cybersecurity/social-engineering), there are many more specific strategies used as well, and we will talk about them in this lesson.

Many of these strategies can be used for both offense and reconnaissance. **Reconnaissance** is the process of interacting with a target in order to gain more information about it.

Reconnaissance is very important in social engineering because *having information about a target makes it much easier for a social engineer to manipulate them*. The rise of social media means there is often no shortage of information about a target available online.



### Social Engineering with Emails

Sending unsolicited emails, also known as **spam**, is a highly effective [social engineering](https://www.codecademy.com/resources/docs/cybersecurity/social-engineering) strategy. Most spam emails that show up in our inboxes are obviously fake, and this is deliberate: The scammers who send these emails want easy victims who won’t realize they’re being scammed. Sure, fewer people will open the email, but those who do open it are more likely to be tricked.

The spam used by Social Engineers is often different from these scammers: it’s meant to be hard to detect in order to slip through spam filters and appear legitimate. Most of us know not to trust emails from random dating sites we didn’t sign up for, but what about emails that appear to come from your organization’s own IT department? These emails often exploit our **trust**, by appearing to come from legitimate sources, and this can be compounded by a technique known as **prepending**.

**Prepending** involves altering the subject line, or attaching a message to the email, that says something like “RE:” or “MAILSAFE:PASSED”, in order to make it appear that:

* We have already been communicating with the sender, OR
* The email has passed a spam filter.

When done correctly, this can make the unsuspecting victim feel an even greater sense of security.

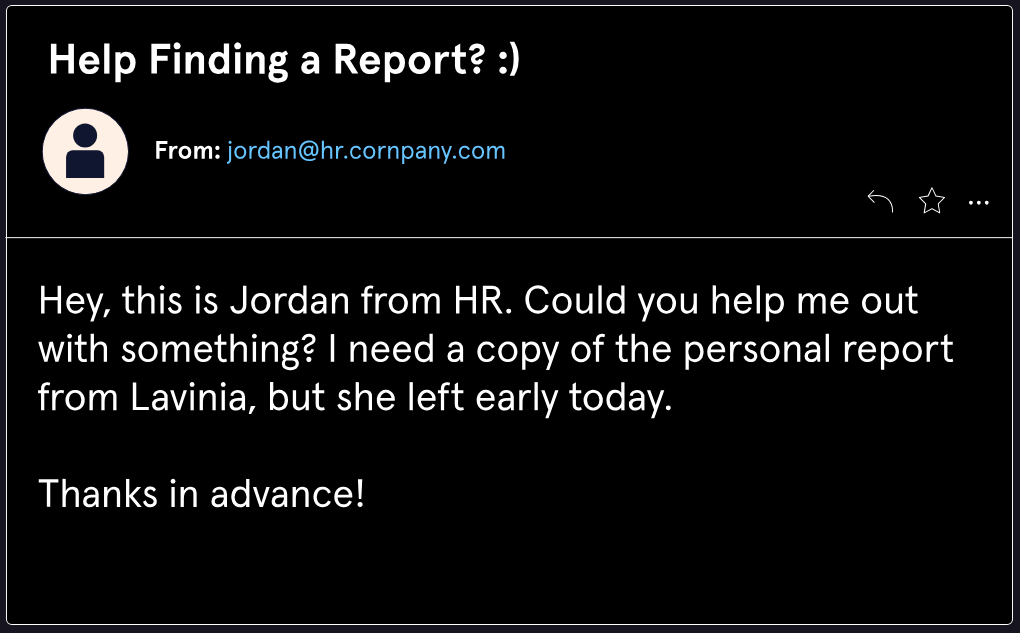


### Hoaxes

Lying to get what you want is a core part of [social engineering](https://www.codecademy.com/resources/docs/cybersecurity/social-engineering), and this can come in many forms. Often, it aligns with one or more of the core principles of social engineering. In social engineering, these lies are often called **hoaxes**. One common hoax is to fake security alerts, creating a sense of **urgency** for the victim. These fake alerts often make use of **trust** and **authority** as well, because the alerts appear to come from real sources which instruct the victim to take the required action.

Another type of lie used in social engineering is **pretexting**, which is when a social engineer invents a false pretext, or reason, for why a victim should share information or carry out an action.

If a random person emailed you asking for sensitive information, you would probably ignore them. On other hand, if the person claimed to be the new point of contact for a contractor working with your organization, you might be fooled into revealing information to them.



### Social Engineering and URLs

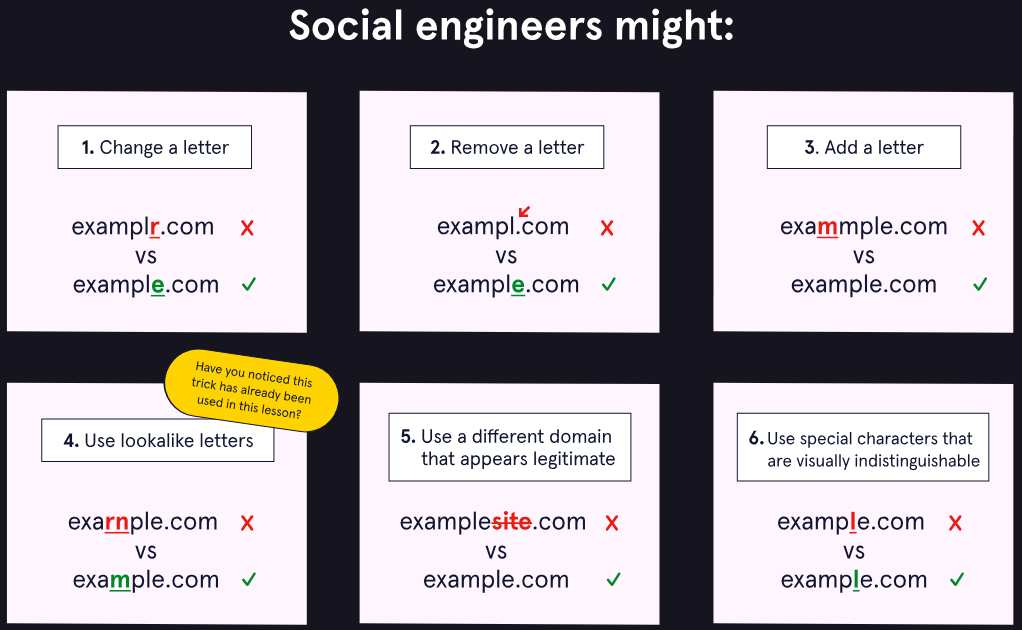
The internet has made it easier than ever to mislead people, and this can occur before a user even connects to a website! Social engineers can trick victims using a simple link to a “trusted” website.

A strategy known as **pharming** refers to when a social engineer redirects victims from a legitimate website to their malicious website instead. This generally involves tampering with DNS information for a computer, a network, or a larger portion of the internet. Pharming can be done by making the name resolution process point to a different IP address. A popular use of this is to harvest banking credentials from unsuspecting victims.

Another strategy used to lure unsuspecting victims onto malicious websites is typosquatting. **Typosquatting** is when an attacker will register a domain very similar to an existing legitimate website, then wait for people to visit the malicious domain. For example, an attacker might register code**A**cademy.com to trick users who are trying to visit codecademy.com. Victims might visit this malicious domain through a mistake as simple as mistyping or misremembering a URL.

If you’re skeptical of this strategy, try finding the differences between these URLs:

* kerning.com vs keming.com
* google.com vs goggle.com



### Identity Fraud

**Identity fraud** refers to when an attacker uses a victim’s personal information. Many of us have heard of examples of malicious actors pretending to be someone for financial gain. For example, using someone else’s credit card or bank account.

Identity fraud isn’t just used for financial gain. It can also be used by social engineers to more convincingly impersonate a victim, either by posing as the victim to mislead others or to gain additional access to the victim’s accounts and resources.

On an individual scale, this could be something like using stolen personal information to “recover” a bank account or target an organization that the individual works for.

This type of fraud can also happen on a larger scale, such as in the form of **invoice scams**, where an attacker alters details on an invoice to steal money. One form of invoice scam involves using [social engineering](https://www.codecademy.com/resources/docs/cybersecurity/social-engineering) to pose as an employee of one organization and sending fraudulent invoices to other companies.

If the attacker is successful, the second company won’t notice the issue and will just pay the invoice!

