How does two-factor authentication protect my users?

Passwords are not strong enough for secure authentication on the web anymore. Even with complicated requirements and heavy hashing, passwords remain vulnerable to common attacks across the Internet. Two-factor authentication adds an extra layer of protection on top of passwords to make logging in safer and protect against many of those common attacks.

But not all two-factor authentication is the same. Most two-factor authentication uses a shared secret to prove the second factor and is still vulnerable to common attacks. Developers have relied on public key cryptography to do secure authentication through systems like SSH for years, and that same cryptography is used in Clef's two-factor authentication to offer users even greater protection.

Clef protects against more threats than any other two-factor authentication system.

This list covers the different vectors of attack that two-factor authentication systems address and the methods for defending against them.

	Brute Force	Bucket Brigade	Keylogging	Server Breach	Stolen Phone	Laptop Theft
Clef	●	●	●	●	●	●
Authy	●		●		●	
Authenticator	●		●		●	
Yubikey	●		●		●	
SMS	●		●			

Brute Force

In a brute-force attack, an individual or a large botnet of infected machines tries to guess millions of credential combinations to gain access to user accounts.

- Every login form can be brute-forced
- Two-factor helps with constant change
- Two-factor does not protect passwords
- Clef signatures cannot be brute-forced

Every login form can be brute-forced

Brute-forcing is the oldest and most obvious way of trying to break into user accounts. To brute force, an attacker guesses from a list of common passwords until they land on the user's. Naive attacks come from a single computer and involve a short list of guesses. These attacks can be blocked by rate limiting login attempts from a certain IP address. If someone guesses too many wrong passwords, the account is locked out and they cannot try any more guesses from that computer.

More sophisticated attackers enlist special hardware[1] and botnets to attack accounts. A botnet is an army of computers across the world that have been infected with malware. An attacker can use a botnet to carry out much bigger brute-forces. Since each computer in the botnet has a different IP address, they can each make several attempts at the password before they are locked out and hand the task off to the next infected machine. Botnets can encompass tens of millions of machines, and as many as 90% of users pick one of the most common 1,000 passwords.[2]

Brute force attacks are the first step for many big attacks, like the one that affected Target.[3] First, the attacker uses brute force tactics to gain access to an account (in Target's case, it was an HVAC contractor's account), then that account is used to upload malware or escalate privileges to gain access to more protected information.

Two-factor helps with constant change

Brute force attacks can only succeed because passwords stay the same, so adding a time-based randomness to the login can make brute-forcing an account much more difficult. In traditional two-factor logins, a user first types their username and password, and then, once that's confirmed, uses a time-based code to prove that they have their phone (or another similar device) in their possession. Since the code is regenerated every 30 seconds, the window for an attacker to brute force it is very small. This second code is called a One Time Password (OTP).

A smaller brute force window is helpful, but many OTP two-factor authentication solutions today do not rate limit the number of attempts that can be made on the OTP. Even with just 30 seconds, an attacker making 10 guesses per second will have a 90% chance of breaking into

[1] http://arstechnica.com/security/2012/08/passwords-under-assault

[2] http://xa.to/1P

[3] http://krebsonsecurity.com/2014/02/target-hackers-broke-in-via-hvac-company

the account over the course of three days.[4]

Two-factor does not protect passwords

Many OTP two-factor implementations also leave password fields vulnerable to brute-forcing, even if they protect the account once the password is compromised. Since the username and password are confirmed before the OTP is ever sent, the attacker can brute force the password without ever being challenged by the second factor. This lets them brute force and learn the password exactly as if two-factor was not enabled. That password can then be used in social engineering attacks[5], OTP brute forcing, or on related user accounts that might not be protected by two-factor.

Clef signatures cannot be brute-forced

Clef uses digital signatures to log users in instead of passwords and OTPs. There is no login form where an attacker could guess Clef credentials — instead we display the Clef Wave which is synced with the user's phone. The phone holds a 2048-bit RSA private key which is used to sign the contents of the wave (along with a timestamp) and send them to the Clef server for verification. This signature is 300 characters long, is specifically tied to the computer where the user is logging in, cannot be used on any other computer, and expires after a few minutes.

While the signature expires over time, the private key that generates it does not, so it must also be safe from brute forcing. The 2048-bit RSA private key is created and stored on the phone. This is the standard key size for RSA public key cryptography today because it would take more than 6.4 billion years to break one.[6]

When users create a new account with Clef, they are prompted to create a four-digit PIN which is used to lock the Clef app on their phone. Four digits only allows for 9,999 different PINs, but the PIN is still almost impossible to brute force. This is because Clef always uses possession of the phone as the primary factor of authentication. The PIN can only be guessed from the phone, so it's not possible for a remote attacker to guess the PIN. If a phone is lost or stolen, the user can deactivate it at getclef.com/lost before the PIN (which is rate-limited) can be brute-forced.

---

[4] http://sakurity.com/otp

[5] http://arstechnica.com/security/2014/09/what-jennifer-lawrence-can-teach-you-about-cloud-security

[6] https://www.digicert.com/TimeTravel/math.htm

Bucket Brigade

(Man-in-the-middle)

In a bucket brigade attack, an attacker intercepts communications between the user and the site where they're logging in.

- Attackers steal secrets in transit
- Two-factor codes can be intercepted
- Clef signatures are tied to a specific computer

Attackers steal secrets in transit

Bucket brigade attacks commonly come in the form of wifi snooping, malicious redirects, or malicious plugins in a user's browser[7]. Once an attacker has intercepted data, they can pretend to be either a user or a site in order to gain access to sensitive account data.

Two-factor codes can be intercepted

In the absence of SSL, traditional forms of two-factor that use one-time passwords are as vulnerable to bucket brigade attacks as passwords. Once communication between two parties is compromised, second factor codes can be read along with passwords. Since two-factor codes are not tied to a specific computer in any way, an attacker who has a stolen password and two-factor code can use them to impersonate a user on a different computer, before the user has a chance to log in themselves. In addition, since attackers can steal passwords through bucket brigade attacks, it's possible that a user's other accounts would also be compromised since password reuse is common.

While some types of traditional two-factor (those which rely on time-based one-time passwords) generate codes which expire quickly, other two-factor methods generate codes which do not change until they are used. Thus, an attack is not necessarily time-sensitive.

Clef signatures are tied to a specific computer

Clef users are protected from this kind of attack because all of the information that is sent between computers, mobile devices, and servers is cryptographically public.

The signature that is used to trigger a login is tied to the specific time and computer where the user is logging in, so it is useless to an attacker who manages to intercept it. Since Clef's private key is stored on the phone and never transmitted between devices, an attacker cannot impersonate a user like they could with a stolen password. In addition, the authorization handshake performed between an integrating website and Clef's server is always encrypted with SSL following the OAuth 2.0 protocol[8].

[7] http://blog.kaspersky.com/man-in-the-middle-attack

[8] http://oauth.net/2

Malware/Keylogging

In a keylogging attack, an attacker installs malware on a computer that tracks what a user types in order to steal passwords or other sensitive information.

- Keyloggers steal passwords
- Two-factor codes are one-time use
- Clef is typing-free so always secure

Keyloggers steal passwords

One of the most common uses of computer viruses is keylogging, where the attacker tracks what a user types and uses the personal information to steal their identity. Because keyloggers grant attackers access to anything a user types, they can often be used to compromise otherwise well-protected accounts. As they've increased in sophistication, keyloggers have even started targeting password manager master passwords[9].

Two-factor codes are one-time use

Traditional two-factor prevents an attacker from gaining access to a user's account because the one-time codes required expire after a single use. With typical keylogging attacks, even if an attacker steals the user's one-time code, because it has already been submitted when the user logged in, the attacker cannot reuse it to gain access and the account stays secure.

Clef is typing-free so always secure

With Clef, no password or other account information needs to be typed, so account credentials are completely protected from this kind of attack.

[9] http://arstechnica.com/security/2014/11/citadel-attackers-aim-to-steal-victims-master-passwords

Server Breach

In order to check whether a password is correct, sites have to store a copy of it on their servers (usually obscured through salting and hashing). In a server breach, an attacker gets access to the stored copy of those credentials in order to compromise a large number of accounts.

- Passwords are symmetrical & vulnerable
- Two-factor systems also symmetrical
- Clef is asymmetrical — nothing valuable to lose

Passwords are symmetrical and vulnerable

In a password architecture, passwords need to be stored in a central database to verify every authentication request. The server becomes a high-value attack target because it stores passwords for all of the site's users. If an attacker gains access to the database, they can compromise every user's account. Attacks like this have made headlines by breaching companies like Evernote[10], Home Depot[11], Twitter[12], and many more.

Two techniques, hashing and salting, have become standard practice for protecting user accounts in the case of a breach[13]. When an attacker breaches the server, they can only see the hash, not the original password. There are several different algorithms that are used to hash passwords. MD5 was the old standard, but slower algorithms like PBKDF2 and bcrypt are now considered best practice.

The slower the algorithm, the longer it takes for attackers to crack the password. It is impossible to turn a password hash into a password, but once an attacker has hashes, they can run common passwords through the hashing algorithm until they guess the right password and get the same hash. Ars Technica showed how an attacker was able to crack 90% of hashes this way in under 20 hours[14].

Since many users have the same password, cracking the password once might match to the hashes of thousands of account, drastically reducing the time it would take to crack an entire database. To avoid this, security best practice is to salt passwords before they are hashed. A salt is a random piece of text which is generated, added to a password, and then stored with it. This means that two users with the same password would each have a different salt added to the end of their password before it was hashed, and so their hashes would look different. This helps slow the process down.

Even when they're hashed and salted, passwords are fundamentally symmetrical. There needs to be one copy in the user's head and another on the server, so they can always be breached through the server. No matter how well salted and hashed, passwords are as strong as users make them.

---

[10] http://www.pcworld.com/article/2030052/evernote-hack-shows-that-passwords-arent-good-enough.html

[11] http://krebsonsecurity.com/tag/home-depot-breach

[12] http://securitywatch.pcmag.com/none/307708-twitter-breached-attackers-stole-250-000-user-data

[13] https://crackstation.net/hashing-security.htm

[14] http://arstechnica.com/security/2013/05/how-crackers-make-minced-meat-out-of-your-passwords

Two-factor systems also symmetrical

Most two-factor authentication systems use One Time Passwords (OTPs) as the second factor for authentication. OTPs are generated by a secret that is shared between the user's phone (or similar device) and the site's server. This secret is called a seed and is combined with the current time and hashed to create the OTP.

The OTP changes regularly, so it is not useful for an attacker to steal. However, since the seed stays the same and must be stored in a readable format on the server, it is vulnerable and valuable. Usually the seed is stored right next to the hashed password and salt, making the two-factor system completely useless in a server breach. Sometimes it is maintained by a separate service or in a separate database, which helps but is ultimately vulnerable to the same kind of attack.

Clef is asymmetrical — nothing valuable to lose

Clef uses asymmetric, public key cryptography to do every authentication. Instead of two copies of the same thing, public key cryptography works more like a key and a lock. Only the lock (the public key) is stored on the Clef server, and it cannot be used to impersonate users if it is ever read by an attacker. The private key is generated and stored on the user's phone, so it can never be lost through a Clef server breach.

This architecture has been trusted by developers using SSH to access remote resources for almost 20 years. Since attackers have to gain access to the physical device with the key, each attack must be targeted at an individual. This severely limits the attack vectors, the economic viability for the attacker, and the responsibility of an authenticating organization.

Lost or Stolen Phone

If a user's phone is stolen, an attacker can use it to attack accounts that are protected by two-factor authentication.

- Lost two-factor codes help attackers
- OTP recovery is nearly impossible
- Clef allows remote deactivation

Lost two-factor codes help attackers

When a phone used for two-factor is lost, it can be used to attack the user's accounts in a variety of different ways. Account lockout attacks prevent the user from accessing the resources they need online to recover their identity or stop the attacker. In the widely published attack on Wired Editor Mat Honan in 2012, the attacker destroyed email accounts and backup services to slow down Honan's recovery of his Twitter handle[15].

Since most email clients (and other important apps) for smartphones keep users logged in forever, attackers can use the phone and the two-factor codes to compromise even the user's best protected accounts.

OTP recovery is nearly impossible

Many two-factor systems that use One Time Passwords (OTPs) require that a user print out and keep a long backup code for every site where they use two-factor authentication. Finding and activating those codes before an attacker can do damage or invalidate the codes is challenging, even if a user is well-prepared.

Backup codes place the burden of security on the user, which result in poor security for most users.

Clef allows remote deactivation

When a Clef user's phone is lost or stolen, they can deactivate it remotely from the Clef website, even if the phone is offline. Clef is not unique in offering this, and Authy has a similar kill switch. To offer remote deactivation, two-factor solutions must have a central identity which is being used across different sites (instead of individual relationships managed through SMS or Authenticator).

[15] http://www.wired.com/2012/11/ff-mat-honan-password-hacker

Stolen Computer

When a user's computer is stolen, any account that they were logged in to at the time of the theft has been immediately compromised, even if they used two-factor to log in. There is no way in most systems for a user to invalidate those existing sessions.

- A stolen computer can easily turn into a stolen identity
- Logging in with two-factor does not help after the fact
- Clef lets users log out everywhere from their phone

A stolen computer can easily turn into a stolen identity

When a computer is stolen, so are all of the accounts which its already logged into. This includes anywhere the user chose 'remember my password' and any site that they had logged in to recently before the theft. A stolen identity can cost a lot more than the laptop, and there's no way for a user to protect their accounts from this kind of theft with passwords.

Logging in with two-factor does not help after the fact

With traditional OTP two-factor, the user is not any safer once they have logged in. If their computer is stolen quickly or the session is alive too long, they lose access to their accounts just as if they used passwords alone. Even though most two-factor authentication ties a login to a user's phone, only Clef keeps that connection to enable logout as well.

Clef lets users log out everywhere from their phone

Since the session is already tied to a user's phone for login, Clef lets a user log out of all of their accounts from their phone. This is convenient in many circumstances, but it can also be a critical security measure when a computer is lost or stolen. Inside the Clef app, the user can tap a button and trigger logouts from all of their accounts remotely. The accounts will not refresh automatically, so any information on-screen can still be viewed by an attacker, but if they try to navigate further or refresh a page, they will find that the account is locked.