# Email as a Master Key: Analyzing Account Recovery in the Wild

Yue Li
College of William and Mary
Williamsburg, VA 23187, USA
yli@cs.wm.edu

Haining Wang
University of Delaware
Newark, DE 19716, USA
hnw@udel.edu

Kun Sun
George Mason University
Fairfax, VA 22030, USA
ksun3@gmu.edu

***Abstract*—Account recovery (usually through a password reset) on many websites has mainly relied on accessibility to a registered email due to its favorable deployability and usability. However, it makes a user's online accounts vulnerable to a single point of failure when the registered email account is compromised. While previous research focuses on strengthening user passwords, the security risk imposed by email-based account recovery has not yet been well studied. In this paper, we investigate the possibility of mounting an email-based account recovery attack. Specifically, we examine the account authentication and recovery protocols in 239 traffic-heavy websites, confirming that most of them use emails for account recovery. We further scrutinize the security policy of major email service providers and show that a significant portion of them take no or marginal effort to protect user email accounts, leaving compromised email accounts readily available for mounting account recovery attacks. Then, we conduct case studies to assess potential losses caused by such attacks. Finally, we propose a lightweight email security enhancement called Secure Email Account Recovery (SEAR) to defend against account recovery attacks as an extra layer of protection to account recovery emails.**

## I. Introduction

Text-based passwords have been used as a dominating solution of user authentication for many decades [31], due to their favorable usability and the fact that they cannot be entirely replaced by other authentication approaches in the foreseeable future [14], [15]. Since text-based passwords are vulnerable to cracking and theft attacks [32], [38], significant research efforts have been made toward enhancing password security from different aspects, including measurement [27], [37], password policy [11], password meters [18], [21], and password managers [30].

Whereas it is critical to secure a password at its creation and input procedures, account recovery as an important component in the entire framework of password-based authentication has been largely overlooked. Account recovery is an irreplaceable link in the password authentication chain. Not being able to provide an easy way to recover the password can cause user frustration, human labor waste, or even user loss. Meanwhile, the account recovery process should also be carefully designed to avoid backdoor threats. Today, most websites rely on accessibility of a registered email of a user to recover or reset forgotten passwords. Though email-based recovery is deployable, compatible, and easy to use, its security implication is understudied. A compromised email account could inevitably become a single-point-of-failure, since an attacker can easily reset the passwords of a victim's other online accounts. Note that such an account recovery attack can naturally circumvent security enhancements on passwords and directly compromise a large number of user accounts by resetting their passwords.

A simple and effective idea is to keep the email account safe. However, this does not happen in a practical world. There is a large number of email accounts leaking to malicious attackers. For example, it was suggested by a security firm in May 2016 [4] that more than 200 million email username/password combinations are in possession of hackers. Major email service providers including Gmail, Hotmail, Yahoo, and Mail.ru are all affected, and millions of email account credentials are compromised. Thus, it is important to understand the security implications of email-based account recovery. A systematic study on its vulnerability, potential damage, and defense has yet to be conducted.

In this paper, we first quantitatively measure the vulnerability of most websites to an account recovery attack. In particular, we manually investigate the account recovery protocols and authentication schemes adopted by the Alexa top 500 websites. We observe that 92.5% of the web services we examined rely on emails to reset user passwords, and in 81.1% of websites, their user accounts can be compromised by solely accessing the registered emails. The difference of 11.4% is due to the lack of username knowledge (i.e., the username/password credential is incomplete) or classifier-based authentication, where abnormal login attempts will be blocked. Afterward, we demonstrate the damage that can be caused by password resets through case studies on four categories of websites, in which we show that significant privacy and financial losses are possible to incur. Then, we exam security policies of eight major email providers. We conclude that a significant portion of leading email service providers fail to take deserved effort to provide user email account protection, leaving them vulnerable to a variety of attack vectors.

Finally, we propose an account recovery protocol named Secure Email Account Recovery (SEAR) as a preliminary solution to address the single-point-of-failure problem of user email accounts. Specifically, the email provider adds an extra layer of protection, which can be in the form of an SMS authentication when a password reset email is intended to be opened. Thereby, the attacker cannot spread the attack by compromising an email account. We demonstrate that SEAR can be easily implemented under the current network infrastructure with full backward-compatibility, and it can strengthen account security with an all-rounded usability consideration (i.e., similar user experience, no need for providing the phone number to all websites that one intended to protect, etc.).

Overall, the major contributions of this work are summarized as follows.

1) We identify the de facto account recovery protocols in the wild by examining the Alexa top 500 websites. In the measurement study, we build taxonomies on websites and account recovery credentials, which enable us to explore the account recovery problem from different perspectives and dimensions.
2) We systematically investigate the email-based account recovery vulnerability that widely exists in today's web services. Our assessment reveals that the security risk is high and could cause severe damage to users.
3) We propose SEAR as a preliminary solution that can be seamlessly integrated into modern email infrastructures in a fully backward-compatible manner. The prototype of SEAR is implemented on open-source mail servers.

The rest of the paper is organized as follows. Section II clarifies related terms. Section III overviews account recovery protocols in modern websites. Section IV evaluates the success rate of an account recovery attack. The extent of damage is presented in Section V. Section VI elaborates SEAR, our preliminary defense mechanism, and its implementation. Section VII surveys closely related works, and finally, Section VIII concludes the paper.

## II. Terminology and Definitions

With the development of new schemes and years of advances in multiple dimensions, the account recovery process cannot be easily elaborated. In order to help understand and organize the heterogeneous makeup of the account recovery process, we first perform a classification on account recovery credentials and websites.

### A. Recovery Primitive, Method, and Protocol

Account recovery is essentially another authentication process, which needs one or multiple legitimacy validations. Each validation is usually done on the server side by matching a mutually agreed-upon piece of credential $\varepsilon$ to the one supplied by the login attempter. While an $\varepsilon$ can be represented by a series of symbols, we categorize $\varepsilon$ into six types based on their sources, as listed below, and we call each type a *recovery primitive* ($\gamma$).

- **Email** ($\gamma_{em}$). Email primitive is the accessibility to a registered email. The validation process may be of various manners, such as sending a hyperlink to reset a password, sending a one-time code for inputting a password reset form, or even directly sending back the original password. Nevertheless, accessibility to the registered email is the only prerequisite.
- **Phone** ($\gamma_{ph}$). Similar to email, phone primitive demands accessibility to a phone that is associated with a pre-registered phone number. The website may choose to call the phone number or send a text message.
- **Security question** ($\gamma_{sq}$). Security question is a kind of knowledge-based authentication, which allows a password reset if questions are answered correctly. Normally, the answers to security questions are intrinsic to users, and hence no extra memory burden is introduced. An example is, "What is your favorite food?"
- **Private information** ($\gamma_{pi}$). Private Information is also knowledge-based authentication in a personally identifiable and thus not massively predictable sense, the answer to which is relatively unique among different users. Although users may or may not intrinsically remember it, they usually have access to the information from other channels. Examples of such information include a credit card number and Social Security Number.
- **Activity Information** ($\gamma_{ai}$). Activity information involves account activity traces. Some service providers believe that a user is expected to be able to recall some of the most basic activities of its account, such as the nickname/username, most login locations, and other users with whom they usually interact. It may even require assistance from acquaintances on the same website.
- **Recovery Token** ($\gamma_{rt}$). A recovery token is usually a non-memorizable piece of information that users possess. Examples are randomly generated tokens at registration or one-time codes generated by mobile applications (authenticators or website-designated apps).

In some cases, websites may ask for a combination of multiple $\gamma$ for stronger authentication and provide multiple such combinations for users to choose from for increased flexibility. To set boundaries among these similar concepts, we define one way to recover a password as a *recovery method*. Fundamentally, a recovery method could consist of one or multiple recovery primitives, and all primitives should be supplied by a user correctly in order to recover its account. For example, on a website $\omega$, one way to recover a user password may be $m_{\omega,1} = \{\gamma_{em}, \gamma_{sq}\}$, meaning that the password can be reset by whomever is in possession of the registered email and answers to the security questions. A recovery method is the most basic unit of a successful account recovery. Similarly, we define the set of all $m$ that a website provides as the *recovery protocol* ($p$) of the website. For instance, for the website $\omega$, its recovery protocol is $p_\omega = \{m_{\omega,1}, m_{\omega,2}, \dots, m_{\omega,i}\}$, indicating that there are $i$ recovery methods and that any recovery method can be used alone to successfully recover an account.

### B. Website Classification

We categorize websites into several groups, helping us look deeper into how different websites handle account recovery in a finer granularity, as well as conduct the damage assessment. Grosse and Upadhyay [23] have done a user account classification based on the values of the accounts. However, their classification is user-oriented, which heavily relies on user-subjective perspective and activity. Namely, different users may have different types of accounts on the same website, depending on the user's purpose for using the websites. By contrast, we take a website-oriented approach by classifying websites based on their service nature. We define the following six website groups with some terminologies acquired from [23].

1) **Routine**. A routine website is one in which users passively receive information. Most of its users produce zero or little long-residing content. Examples of routine accounts are online newspapers, those used for online education, and gaming or music websites.
2) **Spokesman**. Spokesman website accounts usually represent a user's opinion or identity. Users rely on spokesman websites to deliver and exchange information with other real users. Examples of spokesman websites are online social networks, such as Facebook, Yelp, and LinkedIn.
3) **E-commerce**. E-commerce websites mainly involve trading. A business website could be an online retailer, such as Amazon and Ebay, or paid service providers, such as insurance companies. It is common to find addresses, shopping histories, phone numbers, and even payment information in user accounts on these websites.
4) **Financial**. A financial website usually concentrates on financial activities, such as deposits, withdrawals, and online transactions. Examples of financial websites are banking, brokerage, or wallet-type websites, such as Paypal.
5) **Tool**. A tool website does not usually produce a final product. Instead, it provides a tool or platform for helping build or shape the final product. Examples of tool websites are search engines, website builders, online graph drawers, and web traffic analyzers.
6) **Email**. Email websites provide online accounts that are associated with user email addresses, which can send and receive emails, such as Gmail or Outlook.

Nowadays, it is common for websites to have a heterogeneous service nature. It is sometimes hard to classify a website into a single type. For example, Google is a tool website since it offers a search engine. Meanwhile, it is also a spokesman website (Google+) and an email website (Gmail). As such, we sometimes classify a website as multiple types. While allowing such cases, we primarily categorize a website based on its main services and user recognition. For example, an online newspaper may have a review section under an article where users can express and discuss their opinions. However, most users may only browse the news without writing any comment. Thus, the online newspaper is categorized solely as a routine website, instead of a spokesman website.

## III. Account Recovery in the Wild

We manually investigate the account recovery protocols adopted by the Alexa top 500 websites to help understand the protocol composition of modern websites. Since the top 500 websites are ranked by their global web traffic, each of them has a large number of users (or visitors), and thus reflects the de facto techniques adopted for account recovery.

TABLE I: Recovery Primitive Distribution

| Primitive | Number | Percentage | Self-sufficient | Percentage |
|---|---|---|---|---|
| Email | 232 | 97.1% | 213 | 89.1% |
| Phone | 46 | 19.3% | 40 | 16.7% |
| Security Question | 22 | 9.2% | 11 | 4.6% |
| Private Information | 7 | 2.9% | 0 | 0.0% |
| Activity Information | 12 | 5.0% | 10 | 4.2% |
| Recovery Token | 3 | 1.3% | 3 | 1.3% |

"Self-sufficient" implies that the recovery primitive is the sole ingredient in a recovery method (i.e., $|m|= 1$, for example, $m = \{\gamma_{em}\}$).

### A. Demographics

Within the 500 most traffic-heavy websites, we identify 245 websites in which we are able to create an account. Among them, 239 (97.5%) websites have enabled an account recovery protocol ($p$). Since we are only interested in recovery protocols, we consider our dataset to contain only the 239 websites thereafter. There are fewer protocols than websites due to multiple reasons. First, we count the same protocols that share the same database only once, such as all regional Google sites and subsidiaries of Google, like Youtube. This type includes 99 websites. Google alone contributes 55 of them. Second, there are 40 websites that do not have login functionality. For example, some online newsletters do not need user logins. In addition, some recorded sites are just advertisement network referrer links or content delivery networks in which not even an accessible homepage is available. Examples are adnetworkperformance.com and www.t.co. Third, we fail to examine 51 websites with less commonly used languages. It is challenging to recognize and input CAPTCHA in these languages, which is a required process in order to register an account. Finally, on the rest of the websites, a local phone number or membership is mandatory for registration. Examples include most online banking systems. These websites are not open to an outsider, and thus we are unable to access them.

The websites being successfully examined bear a similar distribution on visitor origins in the Alexa top 500 list. Though only a limited number of websites are examined, these popular websites attract most web traffic. For instance, Google alone is reported to account for up to 40% of web traffic [3]. Therefore, we believe that our analysis is representative and can genuinely cover the mainstream of modern website account recovery protocols used by most online users.

Overall, our dataset contains 239 websites that enable account recovery, naturally including 239 password protocols. In these protocols, we identify 324 recovery methods. Then we identify 364 recovery primitives in these recovery methods. On average a website has 1.36 recovery methods, and each method involves 1.12 recovery primitives. This implies that most of the websites provide only one recovery method, and recovery primitives in a recovery method are mostly homogeneous.

Note that nowadays, many websites have used Single Sign On (SSO) for logging in. SSO enables a user to use the account of an identity provider, such as Facebook, to log into other websites. In our dataset, 136 websites feature at least one SSO identity provider. The top three are Facebook (103 occurrences), Google (67 occurrences), and Twitter (35 occurrences). Regarding SSO, users should recover their accounts from the SSO identity provider website, such as Facebook.

### B. *Primitive and Method Usage*

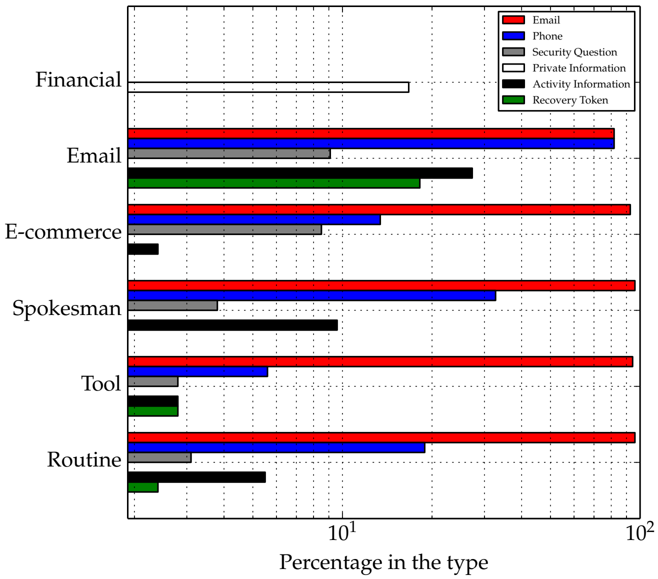
To illustrate the major composition of recovery protocols, we examine the usage of recovery primitives and list the overall occurrence of each primitive in Table I. As shown in the table, using email to recover a password is prevailing: 97.1% of websites include email ($\gamma_{em}$) in their recovery protocols. Furthermore, among 89.1% of websites, email itself is sufficient to recover a password (namely, at least one of their recovery methods contains the element of email primitive only). It is evident that most of the top websites delegate the security responsibility of account recovery to email service providers, instead of extending and relying on their own security infrastructures.

The second most popular method is using a mobile phone, which is seen in a notable portion (19.3%) of websites. Surprisingly, 4.6% of websites still rely exclusively on security questions to recover passwords, which are suggested against by many previous researches [13], [22], [34]. Meanwhile, private information, activity information, and recovery tokens are much less used since they may involve more deployment costs and have privacy concerns. However, these primitives are commonly used in sensitive online services, such as financial institutes.

From Table I, we can also easily infer that most recovery methods contain only one recovery primitive. In fact, recovery protocols in 95% of the websites we examined include at least a single-primitive recovery method. As recovery methods with multiple recovery primitives are rarely found, scattered, and hardly organizable, we focus more on unveiling the structure of a single-primitive recovery method. Following the annotations introduced in Section II-B, we identify 127 routine websites, 82 e-commerce websites, 52 spokesman websites, 36 tool websites, 6 financial websites, and 11 email websites. Their single-primitive recovery methods are portrayed in Figure 1. It is not surprising to see that different genres of websites use different single-primitive recovery methods to balance their own security and usability trade-off.

From the figure, we can also see that financial websites are quite different from the other five — only one website uses a single-primitive method for account recovery (private information). Financial websites are the only type of website that usually has multiple recovery primitives in a recovery method. The other five types of websites all heavily rely on email (more than 80% for email sites and more than 90% for the other four) for account recovery. Using a phone follows as the second most commonly seen account recovery method. Interestingly, email websites themselves heavily rely on a mobile phone to recover passwords and are significantly more prone to use account recovery primitives other than email services. One possible explanation is that the email is already the end point of an account recovery chain and that email service providers prefer not to lead their users to their competitors' email services. Thus, they attempt to offer other remedies, such as a mobile phone.

Fig. 1: Recovery Methods – Single-Primitive



Log scale is used for a clearer presentation of small percentages. The sum of percentage can be more than 100 (with multiple recovery methods in a recovery protocol), or less than 100 (we show only single-primitive recovery methods)

TABLE II: Recovery Methods with Multiple Primitives

| Primitive Number | Routine | Tool | Spokesman | E-commerce | Email | Financial |
|---|---|---|---|---|---|---|
| 2 primitives | 6 | 3 | 3 | 8 | 0 | 6 |
| 3 primitives | 0 | 1 | 0 | 1 | 0 | 5 |

The usage of multiple-primitive recovery methods is not very common, given the fact that on average a recovery method consists only 1.12 recovery primitives. Due to the sparsity, we list the total number of two-primitive and three-primitive methods used in each website category in Table II. None of recovery methods we identified has more than three primitives. Financial and e-commerce websites deploy such multi-primitive recovery methods more than other websites, possibly due to their critical service nature. However, with equal importance, email websites do not have a single recovery method that has more than one primitive.

Overall, it can be inferred that most of the top websites delegate the security responsibility of account recovery to email service providers, instead of extending their own security infrastructures. There could be several reasons for this. First, it keeps high usability, as the registered email address becomes a centralized master key through which users can conveniently manage almost all of their online accounts. In other words, users incur almost no more cost when the number of online accounts increases. Second, email recovery mechanisms can be easily deployed at both sever and client sides. Third, the security obligation of account recovery is delegated to other online services, which significantly reduces the security responsibility of the website. Accordingly, the email of a user ends up as essentially another password manager and thus a potential single point of failure. The illegal access to an email account can pose a serious security threat on most websites, with the only exception being financial websites. As a result, an attacker can easily compromise user accounts by mounting an account recovery attack, especially nowadays when email accounts are at a massive loss [4].

## IV. Attack Assessment

Since emails play a critical role in account recovery, it is necessary to evaluate the vulnerability that may be introduced by emails, especially when user email accounts are at risk.

Although we have observed that most websites rely on emails to recover user passwords, the assumption that possessing a password will compromise a user account may no longer hold under today's multi-dimensional authentication context, where a password may not always be the sole gatekeeper. In fact, an account recovery attack should be considered successful only if an attacker can actually log in to the target website and impersonate the victim user, which rules out cases where the attacker steals a password but fails to access the account due to a lack of other credentials. We first define the capabilities of an attacker and then discuss the possibility of a successful attack.

### *A. Threat Model*

We assume that an attacker has access to the victim's primary email account and attempts to log in to a user account by exploiting the information included in the recovery email. Specifically, the attacker has no knowledge about the victim's personal information and makes no attempt to obtain the information that is believed discoverable or guessable yet hardly quantitatively assessable, such as user-chosen usernames (when it is neither the email address nor included in the recovery email) or security question answers. We also assume that the attacker does not make extra efforts to bypass additional classifier-based authentication schemes, such as IP address or OS/browser fingerprinting. Note that we aim to set a baseline for the success rate of an account recovery attack so that we keep our attack model simple and clean. In the real world, attackers may try to use more sophisticated tactics to break into even more user accounts [9].

### *B. Possibility to Break-in*

As suggested by Table I, 213 out of the 239 websites solely rely on emails to recover user passwords, making 89.12% of the examined websites potentially vulnerable to account recovery attacks. However, an attack may not be successful for two reasons: the lack of other credentials and additional classifier-based authentication. Thus, these factors should also be taken into consideration for estimating the success rate of mounting account recovery attacks on these 213 websites. We discuss the impact of each factor as follows.

TABLE III: Websites Vulnerable to Account Recovery Attacks

| All | $R_1$ | $R_1$&$R_2$ | $R_1$&$R_2$&$R_3$ |
|---|---|---|---|
| 239 | 213 (89.12%) | 201 (84.1%) | 194(81.1%) |

$R_1$: Allows email as an account recovery method. $R_2$: Username is directly obtainable. $R_3$: No classifier is enabled.

*1) Lack of Credentials:* The first factor is the lack of other credentials, and a user's password is not the only credential needed to log in to a system. We investigate the use of a username, as it is also a required piece of information for successful authentication. A website needs to know a username or an email address first to locate an account so that the corresponding recovery methods, such as security questions for the designated user account, can be retrieved. We identify that 80.3% (192 out of 239) of websites allow the use of email addresses as usernames, and 178 of them can use emails to recover their passwords. On the other hand, 23 websites that do not treat email address as a type of username (i.e., a username is freely selected by its user) provide email-based username recovery or directly send a username in the account recovery email, meaning that the username itself can be accessible from the email account. Thus, in total, 84.1% (201 out of 239) of the examined websites are potentially vulnerable to account recovery attacks. In other words, among the 213 websites that allow emails to recover passwords, 12 of them are immune to account recovery attacks because attackers cannot know usernames through emails. Another lack-of-credential scenario is the two-factor authentication (2FA) in which an attacker has no access to the other authentication factor. In this case, the login will also fail. We found that 35 websites feature 2FA options. However, the general adoption rate is still believed to be quite low. By analyzing more than 100,000 Google accounts, Petsas et al. [33] estimated that Google 2FA is adopted by no more than 6.4% of its users in 2015. It is also unlikely for other websites to have a much higher adoption rate than Google. Furthermore, 2FA is an option disabled by default in all of the websites we have identified. Therefore, a 2FA-available website should still be considered vulnerable to account recovery attacks since more than 90% of the users are not really protected.

*2) Classification-based Authentication:* The other factor taken into account is classification-based authentication. Leveraging more or de-centralized credentials may incur significant usability hassles and thus repel users. Therefore, many websites start to use a classifier to automatically verify a legitimate login attempt to balance usability with security, where a correct password is not sufficient for login. The classifier aims to detect anomalous login behaviors by taking many signals into the classification decision, such as the IP address, cookies, and OS/browser fingerprints. Alaca et al. [9] identified and evaluated 29 fingerprinting mechanisms, and each of them may produce multiple signals. If a login attempt is classified as suspicious, the system is likely to trigger a standard 2FA. Authentication classification is reported by Google [8] to effectively reduce 99.7% of account compromises using more than 120 features. However, the classification is a black-box that is hard to comprehend, especially when the features are numerous. To determine whether a website has enabled a classifier, we adopt an attacker-centric approach, where we probe all 239 websites by using the Tor network and VPN[1], which enables us to emulate an attacker. Specifically, we first train each website by manually logging in to the website on the same computer once per day for a period of one week. The computer has a fixed fingerprint and IP address. Then, we camouflage ourselves as a user in a different country with different operating systems and browsers (all cookies cleared) to log in to the same website three weeks after the training stage. Note that we have provided necessary information, which includes a backup email, phone, and security question, for the use of the 2FA to the website when the classifier has low confidence. Our methodology cannot guarantee 100% accuracy of the results since the classification systems of these websites are still unknown. However, we believe that our results are sufficiently close to the ground truth since a useful classifier should capture such obvious anomalies. Our results indicate that only 14 (5.9%) out of the 239 websites are using a classifier, as we are either required to complete a standard 2FA or blocked from logging in. Furthermore, 8 of the 14 websites rank top 30 in web traffic, and the others are mainly financial websites. Clearly, though useful, classification-based authentication has not been widely used, and thus account recovery vulnerabilities still remain, at least at the current stage.

After considering the above two factors, we are able to answer the question of how many websites are vulnerable under such an attack model. We concisely summarize the results in Table III, which shows that overall, 81.1% of the websites we examined are vulnerable under our threat model. In addition, if an attacker is sophisticated and could emulate enough login signals to deceive the classifier, 84.1% of the websites would be vulnerable to account recovery attacks.

## V. Damage Estimation and Email Security

As a large portion of websites are vulnerable to account recovery attacks when a registered email is compromised, we evaluate possible damages that could be caused and the security policies of major email providers, which are essential to throttle attacks on user email accounts.

### *A. Damage*

The damage can be multi-fold. First, attackers are able to steal private information, such as home address and activity history of users. In fact, this is the main reason why an attacker is interested in user passwords. Second, the attacker may also actively impersonate legitimate users to post information, such as sending spam messages on the user's behalf [8]. Third, they may cause financial loss by purchasing products and stealing credit card or bank information. Measuring the extent of the damage can be complex and error-prone since even the same type of websites could have very different user data and security policies.

We estimate the possible losses by examining typical websites from four major website groups, which are routine, tool, spokesman, and e-commerce. We do not examine the email group as Egelman et al. [20] have already done a thorough investigation on how much sensitive information resides in one's primary email account, reporting that a substantial amount of sensitive information can be found in the email archive, such as credit card numbers (16%) and SSN (20%). We also exclude the financial group due to the fact that all of the financial websites we examined in the Alexa top 500 websites are immune to the account recovery attack, as the email is insufficient to reset a password. In our examination, we select those websites with a single service type. In addition, we also try to select these websites that are likely used by normal users. A counter-example is a paid advertisement publisher, which has a high volume of web traffic, but few normal users would use it.

We show the damage assessment in Table IV. It is evident that all of the websites we examined, to various extents, expose user private information, such as phone numbers, birthdates, and addresses, to attackers. In addition to private information, an attacker is able to actively mount subsequent attacks, such as sabotaging or spamming. Sabotaging may not be appealing to the attacker since it does not bring many benefits. However, using a real account for spamming is a common practice among spammers [8]. Furthermore, attackers may even purchase products in online stores with stolen payment information. For the attacker to receive the ordered package or intercept the delivery process, it may need to change the shipping address. We observe that many e-commerce websites require payment authentication, in terms of the credit card security code (Walmart and GAP), to post an order. Amazon requires to re-input the complete payment information if the shipping address is new. However, surprisingly, Ebay allows a user to change the shipping address freely without additional authentication, which makes financial losses largely possible if the account is compromised by attackers.

### *B. Assessing Email Security*

Since email is pivotal to account recovery, the security of user accounts in a website is heavily dependent on the email security. A more secure email service can certainly help to thwart account recovery attacks in the first place.

To this end, we evaluate the security policies of all 11 major email service providers in our dataset, which span different geo-locations, including North America, Asia, and Europe. The fields examined involve several authentication policies, including minimum password length, minimum password composition (uppercase letters, lowercase letters, digits, and special characters), whether 2FA is provided, and whether a classifier is used to filter out abnormal login attempts. The list of providers we examined and results are shown in Table V.

[1] The Tor network is known for having abnormal login issues in some websites, so we use both Tor and VPN to obtain most accurate information.

TABLE IV: Damage Estimation

| | Site | Sensitive Information | Activity | Financial |
|---|---|---|---|---|
| Routine | netflix.com | Phone Number, Watch History, Credit Card Number, Credit Card Info | | Subscribe/Update Service |
| | nytimes.com | Name, Location, Purchase History, Occupation, Income, Gender | | |
| | weather.com | Name, Birthday, Gender, Phone Number, Home Address, Work Address | | |
| | wikia.com | Location, Birthday, Name, Gender, Occupation, Posts | | |
| Tool | github.com | Company, Location, Credit Card Number, Credit Card Info | Sabotage | Purchase Service/Data |
| | dropbox.com | All Files Stored, Access History | Sabotage | |
| | skype.com | Phone Number, Birthday, Location, Connection's Phone Number, Birthday, Location, Gender | | |
| | ebates.com | Name, Address, Shopping History | Spamming | |
| Spokesman | facebook.com | Name, Address, Birthday, Gender, Work, Education, Phone Number,<br>Contact's Information, Posts, Messages | Spamming, Sabotage | |
| | instagram.com | Name, Phone, Gender | Spamming, Sabotage | |
| | reddit.com | Private Messages | Spamming, Sabotage | |
| | quora.com | Private Messages | Spamming | |
| | livejournal.com | Birthday, Location, Private Messages, School, Posts | Spamming | |
| E-commerce | amazon.com | Shopping History, Files on Cloud, Name, Address, Phone Number, Private Messages,<br>Reviews, Browsing History, Credit Card Number, Credit Card Info | | Purchase Products/Services |
| | ebay.com | Name, Gender, Address, Buying History, Selling History, Private Messages, Phone Number, Credit Card Number, Credit Card Info | Sabotage | Purchase Products* |
| | walmart.com | Name, Address, Phone, Shopping History, Credit Card Number, Credit Card Info | | |
| | gap.com | Name, Gender, Address, Phone Number, Shopping History, Credit Card Number Credit Card Info | | |

Red color indicates that the information is fully obtainable while Blue color indicates that the information is only partially obtainable.

* When purchasing a product on Ebay, the user can modify the shipping address without re-inputting payment information.

TABLE V: Examining Major Email Providers

| Provider | Region | Length | Composition | 2FA | Classifier |
|---|---|---|---|---|---|
| Gmail.com | USA | 8 | 1 | √ | √ |
| Yahoo.com | USA | 7-10* | 4-1* | √ | √ |
| Outlook.com | USA | 8 | 2 | √ | √ |
| AOL.com | USA | 8 | 1 | √ | |
| QQ.com | China | 6 | 1 | √ | |
| 163.com | China | 6 | 1 | √ | |
| Sina.com.cn | China | 6 | 1 | | √** |
| China.com | China | 6 | 1 | | |
| China.com.cn | China | 6 | 1 | | |
| Rediff.com | India | 6 | 1 | | |
| Yandex.com | Russia | 8 | 1 | √ | |

* Minimum password length and composition can vary depending on each other. For example, a password of length 7 must have 4 types of characters to be accepted by Yahoo. However, a password of length 10 can have only 1 type of character.

** The on/off of the classifier is configurable, and the default is off.

TABLE VI: Password Policies

| | Routine | Spokesman | E-commerce | Financial | Tool | Email | Overall |
|---|---|---|---|---|---|---|---|
| Length | 5.74 | 5.54 | 6.35 | 7.33 | 5.91 | 7.0 | 5.92 |
| Composition | 1.20 | 1.19 | 1.57 | 1.67 | 1.36 | 1.18 | 1.33 |

For Yahoo.com, we choose a minimum length of 8 and a composition of 2 since this setting may fit well with more typical passwords.

We also list the password policies of all six types of websites in Table VI, with respect to minimum password length and minimum types of characters required (on average). We can see that the minimum length of passwords in email websites is seven, which is only less than that of financial websites. However, email websites have the weakest composition complexity policy, since most of them do not require more than one type of character in a password, and users are more likely to create predictable passwords under such a policy.

We also notice that a significant portion of email providers include 2FA functionalities in their authentication systems. Compared to the overall rate of 2FA-enabled websites, email providers show a much higher security concerns and offer 2FA enhancement to secure user accounts. However, the number of users that actually use 2FA is likely to be small [33]. A more effective solution might be using a classifier to verify a legitimate authentication attempt. Although some of the classification signals can be easily spoofed [9], it is still difficult for an attacker to correctly spoof all signals considered by the classifier, especially when the adopted signals are unknown [15]. Unfortunately, only 4 out of the 11 email providers have integrated such a protection mechanism. One of them (sina.com.cn) requires a user to turn on the classifier, but most users probably do not enable it as the default setting is off. The other 7 email providers are much easier to be compromised by phishing attacks and password guessing/cracking attacks. Under such a condition, those accounts that are associated with a weak email account are vulnerable to account recovery attacks.

Generally speaking, a large portion of major email service providers fail to provide adequate security protection on user email accounts. It makes an account recovery attack more likely to happen, and thus jeopardizes the security of the online accounts that rely on emails for account recovery.

## VI. Securing Email-based Account Recovery

It is not an uncommon scenario that an email provider and its users fail to adequately protect their email accounts. Thus, an email account compromise could trigger massive compromises of other accounts that use emails for recovery. To mitigate account recovery attacks, we propose a lightweight Secure Email Account Recovery (SEAR) protocol that can be seamlessly integrated into current network infrastructures. SEAR does not attempt to change the current email-based account recovery model, but instead, it aims to prevent attackers from recovering other account passwords if an email account is compromised. SEAR requires little effort from a website and its users. In short, SEAR requires a 2FA only when an account recovery email needs to be opened. Meanwhile, the normal email checking experience remains unchanged.

### A. SEAR Specification

The core of SEAR is to add a header in an email to indicate that the email is for account recovery purposes. This method is transparent to existing email infrastructures since Simple Mail Transfer Protocol (SMTP) allows users to customize headers. The basic workflow of SEAR is simple. The account provider (i.e., the website where an account recovery is undergoing) will add a header "tag:value" pair (we choose "Recover:1")

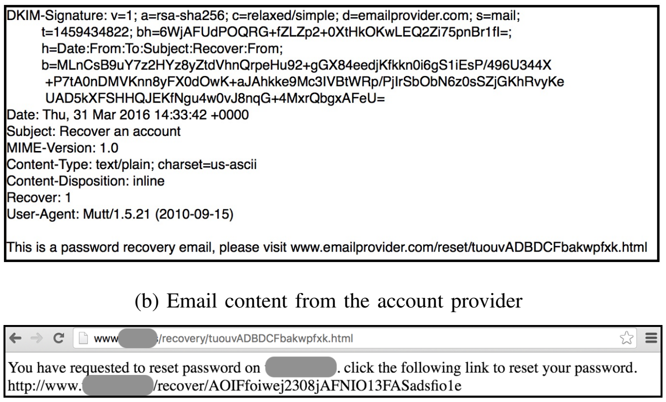


Fig. 2: Account Recovery Examples.

(a) Actual recovery email

```
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=emailprovider.com; s=mail;
        t=1459434822; bh=6WjAFUdPOQRG+fZLZp2+0XtHkOKwLEQ2Zi75pnBr1fl=;
        h=Date:From:To:Subject:Recover:From;
        b=MLnCsB9uY7z2HYz8yZtdVhnQrpeHu92+gGX84eedjKfkkn0i6gS1iEsP/496U344X
         +P7tA0nDMVKnn8yFX0dOwK+aJAhkke9Mc3IVBtWRp/PjIrSbObN6z0sSZjGKhRvyKe
         UAD5kXFSHHQJEKfNgu4w0vJ8nqG+4MxrQbgxAFeU=
Date: Thu, 31 Mar 2016 14:33:42 +0000
Subject: Recover an account
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Disposition: inline
Recover: 1
User-Agent: Mutt/1.5.21 (2010-09-15)

This is a password recovery email, please visit www.emailprovider.com/reset/tuouvADBDCFbakwpfxk.html
```

(b) Email content from the account provider

www[redacted]/recovery/tuouvADBDCFbakwpfxk.html

You have requested to reset password on [redacted]. click the following link to reset your password.
http://www.[redacted]/recover/AOIFfoiwej2308jAFNIO13FASadsfio1e

in the recovery email, and upon receiving an email with the recovery header, the email provider will require the recipient to re-authenticate itself via a second channel in order to access the recovery email's content.

While our solution is straightforward in principle, there are practical challenges that should be carefully addressed. In particular, the current email-fetching protocols (IMAP or POP3) do not protect any specific emails. In other words, the user's mail user agent (MUA), such as Thunderbird, Outlook, and Yahoo! Mail, will fetch whatever emails belonging to the recipient from the Mail Delivery Agent (MDA) of the email provider, without supporting any additional authentication process. Under this emailing infrastructure, it is infeasible to protect any specific emails. To address this challenge, we attempt to protect the recovery email content, instead of the email itself, through the use of an "intermediate token." Specifically, the email provider intercepts the recovery email content and replaces it with an intermediate token. Thus, the recipient will only receive the token. Then, when the recipient opens the recovery email, the intermediate token should be sent back to the email provider by the recipient to initiate the re-authentication. The token submission can be done in different ways, e.g., clicking a URL that embeds the token as in our implementation (see Figure 2a). The email provider will release the content of the original recovery email only if the re-authentication is successful. Furthermore, to ensure the legitimacy of recovery emails, we enforce the use of the Domain Keys Identified Mail (DKIM) [17] protocol, which uses public-key cryptography to ensure the legitimacy of the sender and has already been deployed on more than 80% emails as reported by Google [19].

Overall, SEAR incurs little user experience change since it only requires a second authentication factor in the rare case when a user needs to reset or recover a password. More importantly, it can be seamlessly integrated into today's emailing infrastructures and MDAs. Moreover, it is fully backward-compatible. When one or both email parties do not comply with SEAR, a recovery email will be treated as a normal email.

### B. Implementation

We implement SEAR on well-known open-source projects to demonstrate its simplicity and compatibility. SEAR requires minor modifications on the account provider and email provider sides to meet its specification. We use two Amazon EC2 Ubuntu server instances [1] to act as the account provider and email provider, respectively. They both run Postfix [7], a widely adopted open-source Mail Transfer Agent (MTA). We use Mutt [5], a text-based email agent to generate recovery emails. The legitimacy of emails is protected via openDKIM [6]. On the email provider side, we modify the "clean-up" procedure in Postfix source code to feature recovery header checking and subsequent actions. We use a URL to embed the intermediate token, such that the user can directly click the URL to submit the token to the email provider. A sample of email received by the user is shown in Figure 2a. The page pointed to by the URL requires a second authentication. If successful, the recovery email content is displayed on the webpage, as shown in Figure 2b. The user can then directly reset a password through the URL on this protected page in a conventional manner.

SEAR induces little overhead for two reasons. First, only a small fraction of emails are account recovery emails. By examining different everyday email accounts, we roughly estimate that account recovery emails make up 0.4% of all emails. All other emails are processed normally. Second, its implementation does not introduce any expensive operations. Even in major email providers' distributed systems, it will not introduce any bottleneck. All modules used by SEAR are mature techniques, which have been extensively tested and used for other purposes. The storage overhead is also negligible since the extra data stored are just the intermediate tokens, which can be as small as 20 bytes in our implementation.

## VII. Related Work

Since first being deployed, password authentication has been extensively studied from different aspects, such as its security [28], [32], [38], measurement [12], [29], and enhancement [24], [35], [36]. It is commonly believed that password will continue its dominance in online authentication in the near future [14].

As an important component of password authentication, account recovery becomes increasingly important, especially when users own an increasing number of online accounts. Garfinkel [22] proposed Email-based Identification and Authentication (EBIA), which authenticates users based on their ability to access a certain email address. Although it cannot universally replace password authentication, EBIA has been a primary method of account recovery. Similarly, receiving calls or SMS on cell phones is another de facto recovery scheme. One popular traditional recovery scheme is the use of security questions [26]. However, it has been shown that secret questions are weak in security [13], [34] because the entropy is low, and hence they can be easily cracked through guessing or social engineering. Bonneau et al. [13] also demonstrated that secret questions have a low recall rate and an easily constructible distribution. They uncovered that users try to supply fake answers to make the questions harder to answer, which in turn yields the opposite outcome.

Multi-factor authentication enhances the security of authentication by requiring two or more authentication factors. Although there are cases in which more than two factors are used, such as Bank of America's account recovery [2], 2FA is generally considered as achieving a satisfactory security level. Besides a password, other factors may range from additional knowledge [16] and biometrics [25] to hardware tokens [10]. However, enabling 2FA sacrifices usability since it takes considerably more time and effort to complete the user authentication process. Therefore, nowadays 2FA has a limited adoption rate in practice [33].

## VIII. Conclusion

In this paper, we investigate account recovery at popular websites by examining their recovery protocols. Through extensive analysis of the security features of those websites, we observe that 92.5% of them rely on emails to reset user passwords. Even worse, the user accounts in a significant portion (81.1%) of the websites we reviewed can be easily compromised by mounting an email recovery attack. However, many email service providers fail to realize such security threats and have not yet taken serious actions to protect recovery emails, leading to a single point of failure of using email for account recovery. To mitigate this problem, we introduce a lightweight Secure Email Account Recovery (SEAR) mechanism to provide extra protection on account recovery emails. SEAR can be seamlessly integrated into modern email infrastructures with a full backward-compatibility.

## Acknowledgments

This work is partially supported by the U.S. ONR grants N00014-16-1-3214 and N00014-16-1-3216, and by NSF grant CNS-1618117.

## References

[1] Amazon web service. https://aws.amazon.com/.

[2] Bank of America forgot passcode. https://secure.bankofamerica.com/login/reset/-entry/forgotPwdScreen.go.

[3] Googles downtime caused a 40% drop in global traffic. https://engineering.gosquared.com/googles-downtime-40-drop-in-traffic.

[4] Hold security recovers 272 million stolen credentials from a collector. http://holdsecurity.com/news/the˙collector˙breach/.

[5] The mutt e-mail client. http://www.mutt.org/.

[6] OpenDKIM. http://www.opendkim.org/.

[7] Postfix. http://www.postfix.org/.

[8] An update on our war against account hijackers. https://googleblog.blogspot.com/2013/02/an-update-on-our-war-against-account.html.

[9] Alaca, F., and van Oorschot, P. Device fingerprinting for augmenting web authentication: Classification and analysis of methods. In *Proceedings of the 32nd Annual Conference on Computer Security Applications* (2016), pp. 289–301.

[10] Aloul, F., Zahidi, S., and El-Hajj, W. Two factor authentication using mobile phones. In *IEEE AICCSA* (2009).

[11] Beautement, A., Sasse, M. A., and Wonham, M. The compliance budget: Managing security behaviour in organisations. In *ACM NSPW* (2008).

[12] Bonneau, J. The science of guessing: analyzing an anonymized corpus of 70 million passwords. In *IEEE Security & Privacy* (2012).

[13] Bonneau, J., Bursztein, E., Caron, I., Jackson, R., and Williamson, M. Secrets, lies, and account recovery: Lessons from the use of personal knowledge questions at google. In *WWW* (2015).

[14] Bonneau, J., Herley, C., Van Oorschot, P. C., and Stajano, F. The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In *IEEE Security & Privacy* (2012).

[15] Bonneau, J., Herley, C., van Oorschot, P. C., and Stajano, F. Passwords and the evolution of imperfect authentication. *Communications of the ACM* (2015).

[16] Brainard, J., Juels, A., Rivest, R. L., Szydlo, M., and Yung, M. Fourth-factor authentication: somebody you know. In *ACM CCS* (2006), pp. 168–178.

[17] Crocker, D., Hansen, T., and Kucherawy, M. Domainkeys identified mail (DKIM) signatures. No. RFC 6376. Tech. rep., 2011.

[18] de Carnavalet, X. d. C., and Mannan, M. From very weak to very strong: Analyzing password-strength meters. In *NDSS* (2014).

[19] Durumeric, Z., Adrian, D., Mirian, A., Kasten, J., Bursztein, E., Lidzborski, N., Thomas, K., Eranti, V., Bailey, M., and Halderman, J. A. Neither snow nor rain nor mitm...: An empirical analysis of email delivery security. In *ACM IMC* (2015).

[20] Egelman, S., Jain, S., Portnoff, R. S., Liao, K., Consolvo, S., and Wagner, D. Are you ready to lock? In *ACM CCS* (2014), pp. 750–761.

[21] Egelman, S., Sotirakopoulos, A., Muslukhov, I., Beznosov, K., and Herley, C. Does my password go up to eleven?: The impact of password meters on password selection. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (2013), pp. 2379–2388.

[22] Garfinkel, S. L. Email-based identification and authentication: An alternative to PKI? *IEEE Security & Privacy Magazine* (2003).

[23] Grosse, E., and Upadhyay, M. Authentication at scale. *IEEE Security & Privacy Magazine 11*, 1 (2013), 15–22.

[24] Huh, J. H., Oh, S., Kim, H., Beznosov, K., Mohan, A., and Rajagopalan, S. R. Surpass: System-initiated user-replaceable passwords. In *ACM CCS* (2015).

[25] Jin, A. T. B., Ling, D. N. C., and Goh, A. Biohashing: Two factor authentication featuring fingerprint data and tokenised random number. *Pattern recognition* (2004).

[26] Just, M. Designing and evaluating challenge-question systems. *IEEE Security & Privacy Magazine*, 5 (2004), 32–39.

[27] Kelley, P. G., Komanduri, S., Mazurek, M. L., Shay, R., Vidas, T., Bauer, L., Christin, N., Cranor, L. F., and Lopez, J. Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. In *IEEE Security & Privacy* (2012).

[28] Li, Y., Wang, H., and Sun, K. Personal information in passwords and its security implications. *IEEE Transactions on Information Forensics and Security* (2017).

[29] Li, Z., Han, W., and Xu, W. A large-scale empirical analysis of chinese web passwords. In *Proc. USENIX Security* (2014).

[30] Li, Z., He, W., Akhawe, D., and Song, D. The emperor? s new password manager: Security analysis of web-based password managers. In *USENIX Security* (2014).

[31] Morris, R., and Thompson, K. Password security: A case history. *Communications of the ACM* (1979).

[32] Narayanan, A., and Shmatikov, V. Fast dictionary attacks on passwords using time-space tradeoff. In *ACM CCS* (2005).

[33] Petsas, T., Tsirantonakis, G., Athanasopoulos, E., and Ioannidis, S. Two-factor authentication: Is the world ready?: Quantifying 2FA adoption. In *Proceedings of the Eighth ACM European Workshop on System Security* (2015).

[34] Schechter, S., Egelman, S., et al. It's no secret. Measuring the security and reliability of authentication via secret questions. In *IEEE Security & Privacy* (2009).

[35] Sun, H., Sun, K., Wang, Y., and Jing, J. TrustOTP: Transforming smartphones into secure one-time password tokens. In *ACM CCS* (2015).

[36] Wang, L., Li, Y., and Sun, K. Amnesia: A bilateral generative password manager. In *IEEE ICDCS* (2016).

[37] Weir, M., Aggarwal, S., Collins, M., and Stern, H. Testing metrics for password creation policies by attacking large sets of revealed passwords. In *ACM CCS* (2010).

[38] Weir, M., Aggarwal, S., De Medeiros, B., and Glodek, B. Password cracking using probabilistic context-free grammars. In *IEEE Security & Privacy* (2009).