Bits, Bytes, and Breaking News

The Hacker News

Home Newsletter Webinars

The $10 Cyber Threat Responsible for the Biggest Breaches of 2024

Jan 16, 2025 The Hacker News

```
{
    "source_type": "Telegram",
    "breach_type": "Stealer Malware Logs",
    "app_url": "https://myesluwsb2cp01.b2clogin.com/
myesluwsb2cp01.onmicrosoft.com/oauth2/v2.0/authorize",
    "breach_publication_date": "2024-05-13",
    "breach_description": "The following credentials
were obtained using a stealer malware, and shared on
the Telegram channel \"GgBest FYRIeIe COMBO @ggbestc\"
(channel id: 2120162084) on 2024-05-13. They were
extracted from a file named \"Shopping Worldwide
combos (7).txt\", with a size of 59.2 MB.",
    "post_title": "Daily Telegram Group DB-Markeet "
```

You can tell the story of the current state of stolen credential-based attacks in three numbers:

- Stolen credentials were the **#1 attacker action** in 2023/24, and the breach vector for **80% of web app attacks**. (Source: Verizon).
- Cybersecurity budgets grew again in 2024, with organizations now spending almost **$1,100 per user** (Source: Forrester).
- Stolen credentials on criminal forums cost **as little as $10** (Source: Verizon).

Something doesn't add up. So, what's going on?

In this article, we'll cover:

- What's contributing to the huge rise in account compromises linked to stolen creds and why existing approaches aren't working.
- The world of murky intelligence on stolen credentials, and how to cut through the noise to find the true positives.
- Recommendations for security teams to stop attackers from using stolen creds to achieve account takeover.

Stolen credential-based attacks are on the rise

There's clear evidence that identity attacks are now the #1 cyber threat facing organizations. The attacks on Snowflake customers in 2024 collectively constituted the biggest cyber security event of the year in terms of the number of organizations and individuals affected (at least, if you exclu CrowdStrike causing a worldwide outage in July) — certainly, it was the largest perpetrated by a criminal group against commercial enterprises. It has been touted by some news outlets as "one of the biggest breaches ever."

Around 165 organizations using Snowflake (a cloud-based data warehousing and analytics platform) were targeted using stolen credentials harvested from infostealer infections dating as far back as 2020. These affected accounts also lacked MFA, enabling attackers to log in with a single compromised factor.

The impact was massive. In all, 9 victims have been named publicly following the breach, impacting hundreds of millions of people's sensitive data. At least one victim paid an undisclosed ransom fee.

But this wasn't a one-off. These attacks were happening constantly throughout 2024.

- The huge Change Healthcare breach, which culminated in 100 million customers being impacted and a $22 million ransom demand, started with stolen Citrix credentials.
- Disney's Confluence servers and Slack instance were hacked, resulting in huge amounts of commercially sensitive data and IT infrastructure details being leaked, as well as messages from 10,000 Slack channels.
- Microsoft suffered a significant breach of their Office 365 environment, with sensitive emails leaked after a "test" OAuth application was compromised using stolen creds.

- Finastra, Schneider Electric, Nidec, Foundation, ADT, HealthEquity, Park'N Fly, Roku, LA County Health Services, and many more all suffered data breaches of varying severity as a result of stolen creds.

Researchers are getting in on the action too. In October, Microsoft's ServiceNow tenant was hacked using stolen credentials acquired online, accessing thousands of support ticket descriptions and attachments, and 250k+ employee emails.

Stolen credentials are still a problem? Really?

Key to many of the attacks targeting workforce identities and online accounts is the use of stolen credentials. And unfortunately, an increased focus on MFA adoption hasn't quite solved the problem.

- MFA gaps remain rife. Research from Push Security shows that where a password is the sole login method for an account, these accounts lack MFA in 4 out of 5 cases.
- The number of breached credentials continues to grow at an alarming rate due to the prevalence of infostealer compromises. And data breaches tend to beget more data breaches as account information is leaked, creating a vicious cycle.
- The shift to third-party apps and services for most major business operations, leading to more accounts, more credentials, and more valuable business data in the cloud — all low-hanging targets for attackers.

So, there are more targets for attackers, more credentials to use against them, and MFA (in particular phishing-resistant MFA) is nowhere near as present as we'd hope. Look at the breaches we mentioned earlier — many of the victims are huge companies, with vast security budgets. If they can't achieve complete coverage, then how can anyone be expected to?

The rise of infostealers

The rise of infostealer malware has had a significant impact on the increase in credential-based attacks.

While infostealer malware isn't exactly new, it's a growing concern for many security organizations. Commercial Malware-as-a-Service offerings on the criminal underground are being continuously updated to evade detection controls, and the more sophisticated criminal and nation state-backed threat groups are proficient in creating custom malware. It's a cat-and-mouse game, and the sheer number of compromised credentials tracing back to infostealer infections is a testament to their success.

Once stolen, credential data such as usernames, passwords, and session cookies makes its way to criminal forums on both the clearweb and the darkweb. Popular infostealers even have their own dedicated Telegram channels to advertise and sell stolen data.

But the landscape in which they are deployed has evolved too. There's a greater appetite for stolen credentials among cyber criminals, and ultimately the more apps that companies use (typically 200+ for the average organization), the more accounts they have connected to them, and the more credentials there are to steal. And because infostealers target all credentials saved on the victim's device (not just those belonging to a single app/website as per phishing campaigns) they're perfectly poised to smash and grab.

Modern working arrangements open up the attack surface further. All it takes is for a user to log into their personal browser profile on a corporate device (or the inverse), and their personal device to be compromised, for corporate credentials to be stolen. And because infostealers are pushed through unorthodox channels compared to more traditional email-based attacks (like gaming forums, Facebook ads, and YouTube video descriptions) it's no surprise that unsuspecting victims are falling foul.

And with password reuse incredibly common (10% of accounts have a breached, weak, or reused password and no MFA), stolen credentials from personal accounts can often be used to access corporate apps too. All it takes is an attacker with a little patience — or the skill to automate SaaS credential stuffing at scale.

The modern identity attack landscape has changed (a lot)

In the past, security and IT teams were masters of their own Active Directory universe, making it possible to participate in password-cracking exercises or to compare threat intel lists to passwords in use by employees.

That picture has changed. Security teams now face a tangle of managed and unmanaged SaaS as critical business operations have moved online. They lack visibility into identity posture on these apps, and the vast majority of organizations do not even have a plausible method for identifying all their accounts and apps in use across the business.

SaaS attack paths leave little room for error

Identity attacks are now fundamentally different. Unlike traditional network-based attacks, attacks that target online accounts follow a much more direct attack path.

Traditional attacks progress by network access, lateral movement, privilege escalation, and other familiar activities. These kinds of attacks are well understood by security teams and existing tooling can observe and detect these techniques.

But account takeover requires an attacker only to compromise an account (the point of initial access) from where they can collect and exfiltrate data from the compromised app. The attack can be over very quickly, and traditional tooling offers little to prevent malicious activity in-app.

Given the weak state of SaaS logging, it's likely that most app compromises won't even be visible to the security team. Even if data is available, detection and response becomes much more difficult after account takeover. There is limited log data available from SaaS to begin with, and distinguishing legitimate user activity from malicious activity is difficult.

We saw with the Snowflake breaches that attackers simply logged in to user accounts using stolen credentials and then used a utility to perform account takeover and recon at scale, ending by using SQL commands to stage and exfiltrate data across multiple Snowflake customer tenants.

Response activities are also constrained by circumstances: Do you have admin rights to the app? Does the app provide the kinds of response activities, such as forcing a session logout, that you need to perform?

Each incident can feel like a one-off investigation, with peculiarities in each app to identify and work through, and few opportunities to automate security responses – limiting response teams to postmortem activities, who find themselves unable to contain or reduce the scope of the breach.

What about threat intelligence?

Threat intelligence on stolen credentials is plentiful — many commercially available feeds can be acquired and ingested by security teams. However, the challenge is finding out where these creds are actually being used, and separating out the false positives.

Researchers at Push Security recently evaluated threat intelligence data representing 5,763 username and password combinations that matched domains in use by Push customers. They found that fewer than 1% of the credentials in the multi-vendor dataset were true positives — meaning that the suspected stolen credentials were still in use by employees at those organizations.

In other words, 99.5% of the stolen credentials they checked were false positives at the time of review.

To deliver on the promise of threat intelligence in a meaningful way, security teams need a different approach. For a start, they need to be able to securely observe and match the passwords found in credential feeds with those being used.

Most organizations fail to extract much value from compromised credential feeds. At most, you might be automating the process of requesting that users check their credentials for their primar SSO login (e.g. Okta, Entra, Google Workspace) when a credential breach notification comes through. But this workflow won't scale when you consider how often these breached credential lists are recycled — it all starts to get a bit spammy. After a while, users will start to complain and ignore these requests.

How security teams can prevent account takeover from stolen credentials using browser telemetry

Security teams need a modern approach to defending against account takeover by preventing stolen credentials from being used, and MFA gaps being exploited.

Push Security provides a browser-based ITDR platform that deploys a browser agent to employee browsers in order to stop identity attacks.

Push uses a browser agent that is able to securely observe credentials at the time of login to any app, in addition to collecting rich browser telemetry and providing security controls designed to stop account takeovers before they occur.

Push is also able to supply browser telemetry and an inventory of your entire identity attack surface of accounts and apps, as well as analyze the security posture of employee passwords, login methods, and MFA status — to close off high-risk account vulnerabilities.

Push recently released two capabilities geared toward helping security teams stop account takeovers caused by stolen credentials and MFA gaps.

Correlate the credentials your employees use with those found in compromised credential feeds

The Push browser agent is able to compare suspected stolen credentials supplied by TI feeds to creds actually in use by employees across your organization and then flag only the verified true positives.

Push customers can consume TI from the sources supplied directly by the Push platform — or use the Push REST API to submit their own email/password combos from existing TI tools.

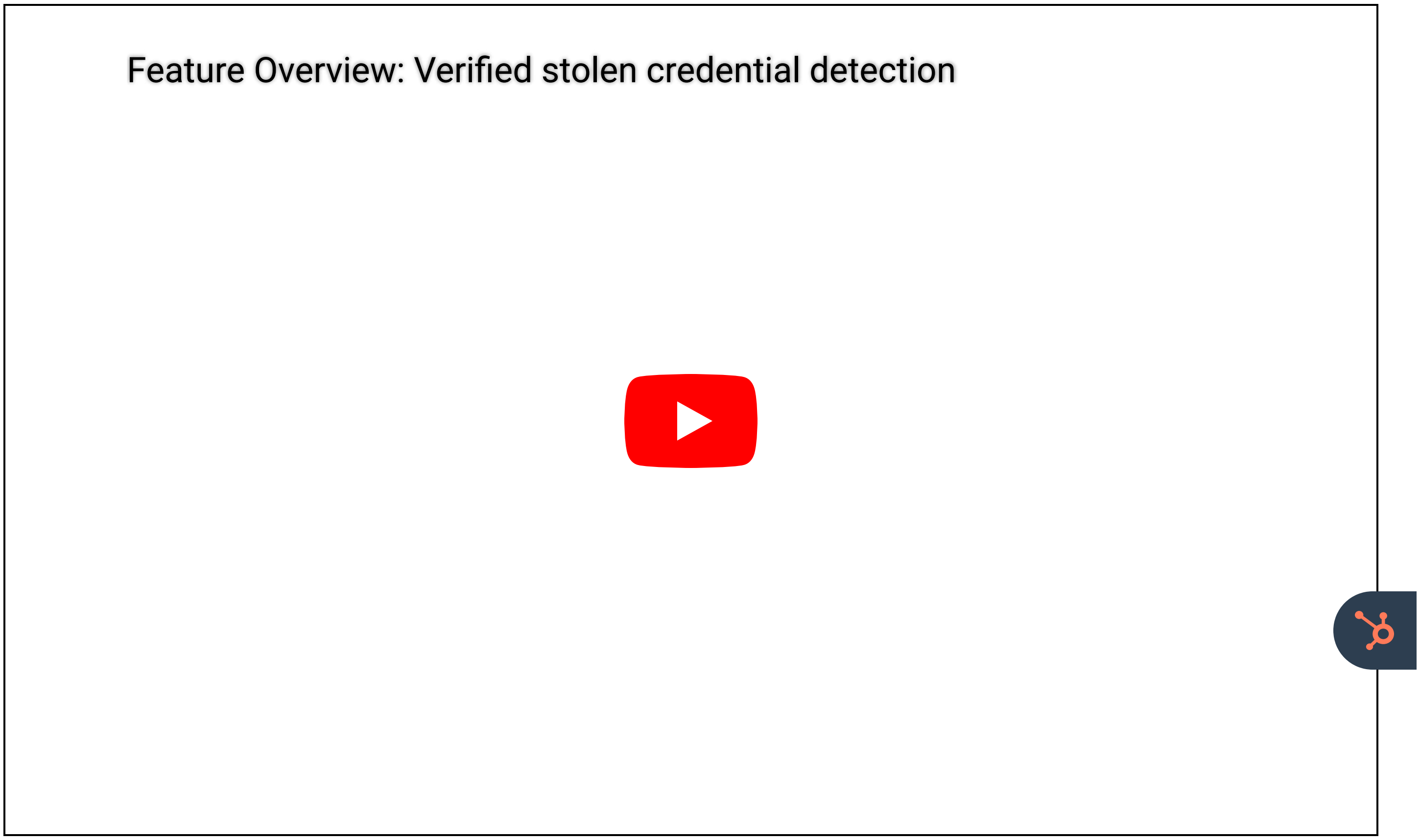
This method works regardless of the source of the data or its age. This method also uncovers where a stolen credential on one app is also in use on several other apps.

Here's how it works:

- Push receives TI on stolen credentials from vendor feeds.
- For each customer environment, Push checks for customer domains in the data set.
- When suspected stolen creds for a customer environment are present, Push hashes and salts the passwords and then sends those fingerprints to the relevant browser agents for comparison. For customer-supplied credential data, Push performs the same salting and hashing to create fingerprints it can use to compare to password fingerprints observed by the relevant browser agents.
- If the stolen credential fingerprint matches a known credential fingerprint observed to be in use by the Push browser agent, the platform returns a validated true positive alert.

You can receive alerts for this detection via webhook, messaging platform notification, or in the Push admin console.

Check out the feature release video for more information below:



Get MFA visibility across all your apps and close the gaps

Push can also help teams close MFA gaps. As users access apps with their corporate identities, Push analyzes their MFA registration status and methods, and also identifies which apps they're using and their login methods. Using in-browser controls, Push can guide users to register MFA across different apps.

Imagine a scenario where you need to quickly investigate the business impact of a recently announced SaaS breach. Using Push, you can:

- Immediately check whether the Push extension has observed employee usage of the breached app. You can also see how many accounts Push has seen on that app and how they are accessing it (SSO vs. other methods, such as local password login).
- For those accounts on the breached app, you can quickly see whether they have MFA, and which methods are registered. To determine MFA status, the Push extension uses the existing user's active session on an app to query that account's MFA registration status using the app's own API, providing a trustworthy verification.
- You can also see whether the users' passwords have any security issues, such as a verified stolen credential, or a password that's weak or reused.

- For accounts that lack MFA, you can then configure an enforcement control to prompt employees who lack MFA to set it up whenever they next use the app.
- Then, use Push's webhooks to monitor for MFA registrations and password changes by querying browser telemetry supplied by the Push agent.

You can learn more about this feature here.

By combining alerting for verified stolen credentials with the ability to find and increase MFA adoption even on unmanaged apps, Push offers security teams a formidable toolkit for stopping account takeover.

Find out more

If you want to learn more about identity attacks and how to stop them, check out Push Security — you can **try out their browser-based agent** for free.

Found this article interesting? This article is a contributed piece from one of our valued partners. Follow us on **Twitter** and **LinkedIn** to read more exclusive content we post.

Tweet Share Share