Fighting Back Against Compromised Credentials

A SANS First Look

Written by **Jake Williams**

Introduction

SANS took a look at the Enzoic for Active Directory product for detecting credential compromise and preventing account takeover. This First Look paper will detail some of the features and our first impressions of the product. Before discussing the Enzoic product, however, it's worth highlighting the problem it solves.

Unfortunately, employees still choose phenomenally bad passwords that are often found in lists of compromised passwords. No matter how many times we tell users not to use the same password for Active Directory and external sites with weak security, they continue to do so.

Auditing passwords is a potential control, but it's a reactive activity.

Auditing passwords is a potential control, but it's a reactive activity. Organizations need a security control that prevents users from choosing bad passwords (including those previously compromised) in the first place. But even this check represents a point in time. If a user sets a password that shows up on compromised lists days later, the organization's security team needs to know that. And the impact of a compromised password is nothing compared to the impact of a compromised credential pair (username and password combination).

Even armed with the knowledge that credentials have been compromised, how should the team react? It's natural to arrive at a knee-jerk reaction of "disable the account," but even such a seemingly simple response ignores the reality that with many users working remotely, one does not simply disable an account. Flexible response options are needed for today's workforce and, to the extent they can be automated, all the better.

To be maximally effective, any modern password security tool must:

- Prevent users from reusing a previously compromised password
- Continually audit passwords in use to determine if they've become compromised
- Notify the security team if passwords have been compromised
- Differentiate between password and credential compromises
- Allow the security team automated response options
- Support response options that can be used with remote workers

The First Look

In our evaluation of Enzoic, we found that it supports all of the must-haves that typical security teams would want. Active Directory is (and always has been) rather limited in how it supports enforcing password complexity requirements and doesn't support modern requirements for a compromised passwords blacklist. Password filtering **DLL**s have long been used to support additional filtering restrictions. Enzoic certainly isn't the first to perform password filtering offering custom dictionaries. And if that's all it offered, we wouldn't bother writing this paper.

What's different about Enzoic is that it securely checks against a proprietary database of compromised passwords and full credentials at both password reset and continuously thereafter. For this checking not to intrude on the user's time, it must be nearly instantaneous—measured in milliseconds rather than minutes.

As a security practitioner, one of my biggest fears is breaking the business. And when I hear about a solution that's checking passwords at the time they're set, I immediately wonder:

- Does it fail open or closed?
- What's the security impact of password checking?
- How are hashes checked securely?

The good news here is that Enzoic fails open and with almost zero impact. Failing open is important because we need users to be able to update passwords, even if checks can't be performed in real time for some reason (a network outage, for instance). While the "near zero impact" claim seems a stretch, it is actually a reality. Because Enzoic checks passwords at the time they're set, it has the hash of the password. Even in ordinary circumstances, the hash is checked against newly compromised password lists on a regular basis. Because the password check isn't just a point in time, passwords can be identified as compromised long after they are set. And that's one of the killer features of Enzoic: Passwords are regularly audited against a continuously updated list of those compromised passwords and full credentials sourced by Enzoic's dedicated threat research team.

Enzoic also realizes that the confidentiality of customers' passwords is and always will be a concern. Simply sending client hashes to an online check would expose those hashes to Enzoic. This is why Enzoic uses the K-Anonymity security model where only a segment of a hash is queried to Enzoic. Enzoic then returns full candidate hash matches that are checked for matches in the client environment. This ensures (rightly) that full hashes never leave the client environment.

Enzoic regularly audits password hashes against a continuously updated dataset sourced by the company's dedicated threat research team.

There have been other solutions that perform password checks, but there's usually a compromise to be had. The check performed in real time (if real-time checking is supported at all) is against a list of limited size, with checks against the full list (whatever that means) occurring offline. This procedure increases user friction substantially because the user may only find out long after the fact that the password they chose was unsuitable for use.

Enzoic provides flexible features for a response depending on a credential pair or only a password is discovered as compromised. This is a risk-focused approach to both detection and remediation. Obviously, a compromised credential pair (username and password combination) represents a higher risk than just a password. Granular response options provided by Enzoic allow organizations to respond in accordance with their risk tolerance.

Enzoic also features root password detection. This feature prevents users from simply iterating or appending numbers and symbols on an unsafe password when it's time to change ("password1" becomes "password2" for example). Threat actors know that if a compromised credential pair no longer works, there's a high chance that it's simply been iterated on, especially if it starts or ends with a number. Using Enzoic robs threat actors of this useful technique.

From a risk perspective, all users aren't equal. Enzoic understands this and supports the creation of user groups to which different policies can be applied. For example, the risk of a regular user credential compromise is completely different from that of a privileged user. You might want to create more restrictive policies for contractors or less restrictive policies for organizationally sensitive users such as doctors or lawyers.

When Enzoic detects compromised passwords, it offers extremely flexible response options that should be appreciated by teams of any size. Enzoic can:

- Force a password change on the next login (with optional configurable delay)
- Disable the account (with optional configurable delay)
- Notify only

The configurable delay is critical, especially with employees working remotely. Disabling an account has always been a high-friction response action, but there's a substantial difference with remote workers. A remote worker can't walk to the help desk to re-enable their account. Re-enabling an account remotely, a practice security personnel should ideally avoid, opens the door to social engineering attacks. By supporting configurable delays, security can meet operational needs with a risk-appropriate model whereby the user is notified of a compromise and, if they fail to take action within the delay period, the prescribed action is taken. Security teams today are all about automation and the Enzoic solution supports automation in both detection and response.

While this First Look paper does not afford the space to cover each of these features in depth, we do want to highlight a few additional aspects:

- Installation was easy. Enzoic is a product for which organizations can easily establish proof of value (POV) because it doesn't require substantial effort to deploy. Removal (not that you'd want to) is also easy.
- A Windows client is available, so when a user picks a blocked password, they're told why the password can't be used. Other password filtering solutions block passwords without explaining why or without checking a complete database, increasing friction of adoption.
- The Enzoic solution integrates with Azure-AD as long as password writeback is enabled.
- Password policies are extremely granular and easy to configure. In a longer product review, this feature would merit a whole section unto itself!
- Domain controllers may not be allowed to talk to the internet, so Enzoic supports proxies to query online data sources for compromised passwords.

Conclusion

Overall, we found Enzoic to be an extremely capable product. Its features demonstrate that the designers and developers behind it understand how security teams operate. Seasoned security professionals can look at a product and see that it was designed with security teams in mind—and Enzoic definitely fits that bill. If you're looking to protect your organization from credential compromise, Enzoic is a tool your organization should investigate.

SANS would like to thank this paper's sponsor:

ENZOIC