Evolved phishing: Device registration trick adds to phishers' toolbox for victims without MFA | Microsoft Security Blog

By Microsoft Threat Intelligence

Published: 2022-01-26 · Archived: 2026-04-05 16:43:25 UTC

We have recently uncovered a large-scale, multi-phase campaign that adds a novel technique to traditional phishing tactics by joining an attacker-operated device to an organization's network to further propagate the campaign. We observed that the second stage of the campaign was successful against victims that did not implement multifactor authentication (MFA), an essential pillar of identity security. Without additional protective measures such as MFA, the attack takes advantage of the concept of bring-your-own-device (BYOD) via the ability to register a device using freshly stolen credentials.

The first campaign phase involved stealing credentials in target organizations located predominantly in Australia, Singapore, Indonesia, and Thailand. Stolen credentials were then leveraged in the second phase, in which attackers used compromised accounts to expand their foothold within the organization via lateral phishing as well as beyond the network via outbound spam.

Connecting an attacker-controlled device to the network allowed the attackers to covertly propagate the attack and move laterally throughout the targeted network. While in this case device registration was used for further phishing attacks, leveraging device registration is on the rise as other use cases have been observed. Moreover, the immediate availability of pen testing tools, designed to facilitate this technique, will only expand its usage across other actors in the future.

MFA, which prevents attackers from being able to use stolen credentials to gain access to devices or networks, foiled the campaign for most targets. For organizations that did not have MFA enabled, however, the attack progressed.

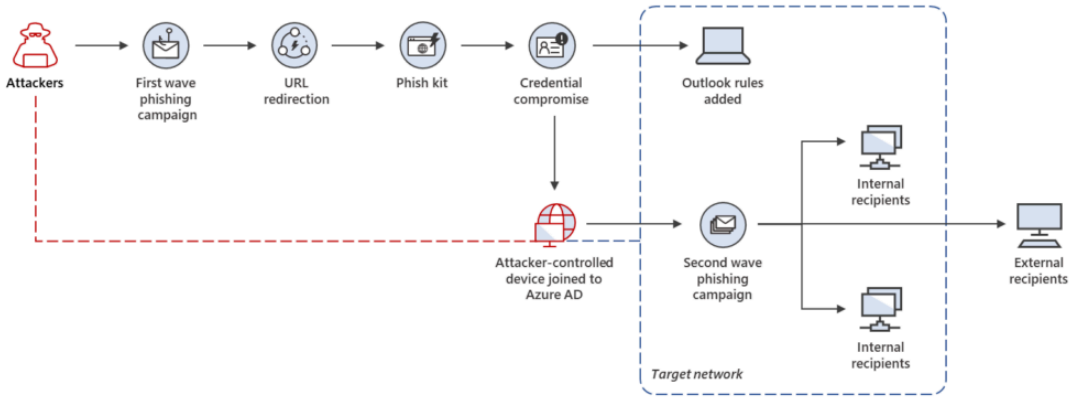

Figure 1. Multi-phase phishing attack chain

Phishing continues to be the most dominant means for attacking enterprises to gain initial entry. This campaign shows that the continuous improvement of visibility and protections on managed devices has forced attackers to explore alternative avenues. The potential attack surface is further broadened by the increase in employees who work-from-home which shifts the boundaries between internal and external corporate networks. Attackers deploy various tactics to target organizational issues inherent with hybrid work, human error, and "shadow IT" or unmanaged apps, services, devices, and other infrastructure operating outside standard policies.

These unmanaged devices are often ignored or missed by security teams at join time, making them lucrative targets for compromising, quietly performing lateral movements, jumping network boundaries, and achieving persistence for the sake of launching broader attacks. Even more concerning, as our researchers uncovered in this case, is when attackers manage to successfully connect a device that they fully operate and is in their complete control.

To fend off the increasing sophistication of attacks as exemplified by this attack, organizations need solutions that deliver and correlate threat data from email, identities, cloud, and endpoints. Microsoft 365 Defender coordinates protection across these domains, automatically finding links between signals to provide comprehensive defense. Through this cross-domain visibility, we were able to uncover this campaign. We detected the anomalous creation of inbox rules, traced it back to an initial wave of phishing campaign, and correlated data to expose the attackers' next steps, namely device registration and the subsequent phishing campaign.

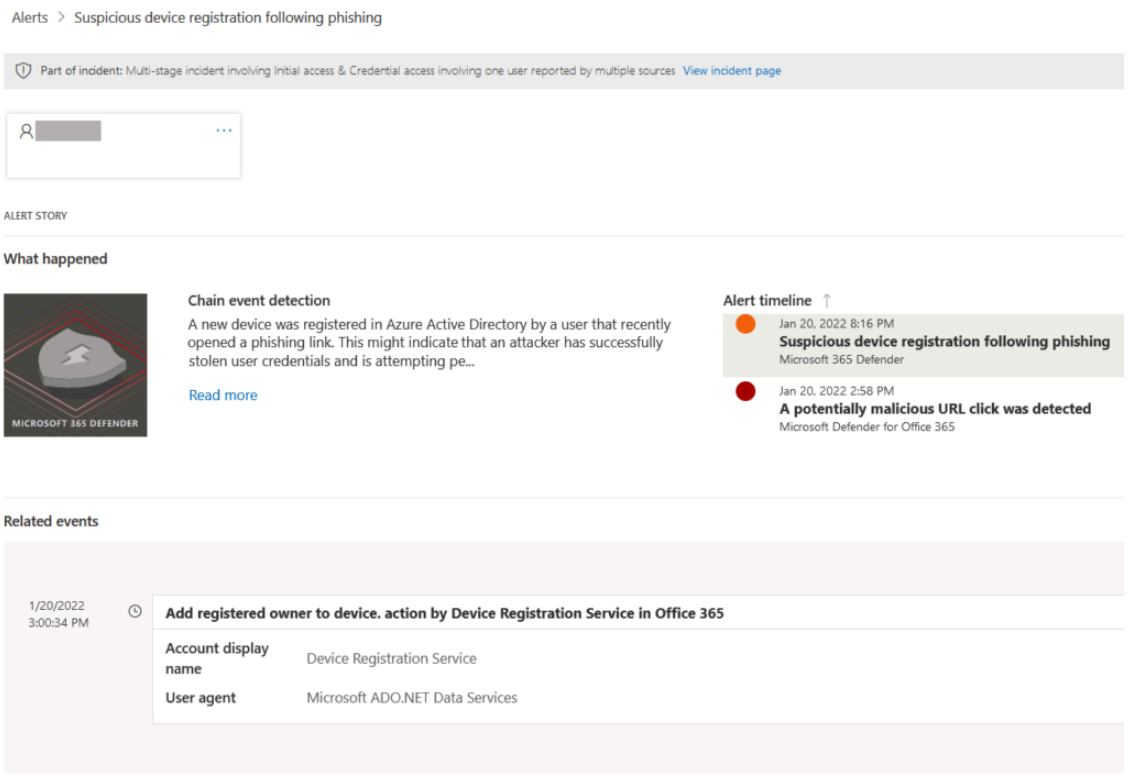

Figure 2. Microsoft 365 Defender alert "Suspicious device registration following phishing"

This attack shows the impact of an attacker-controlled unmanaged device that may become part of a network when credentials are stolen and Zero Trust policies are not in place. Microsoft Defender for Endpoint provides a device discovery capability that helps organizations to find certain unmanaged devices operated by attackers

whenever they start having network interactions with servers and other managed devices. Once discovered and onboarded, these devices can then be remediated and protected.

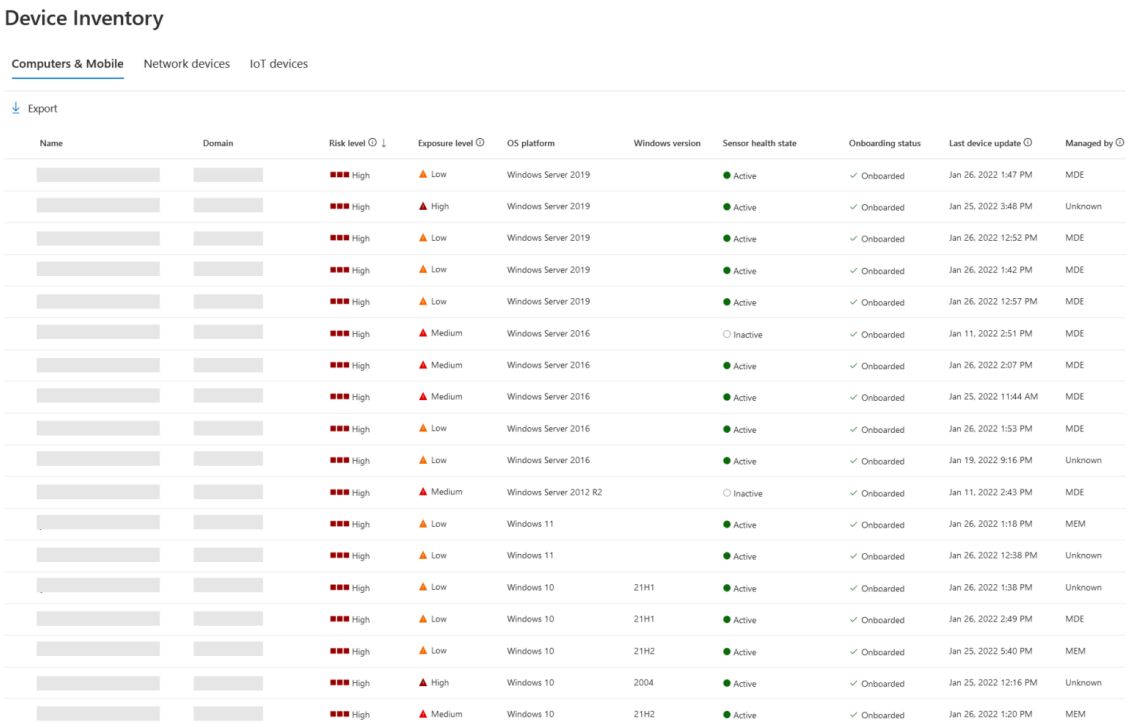

Figure 3. Microsoft Defender for Endpoint device discovery

In this blog post, we share the technical aspects of a large-scale, multi-phase phishing campaign. We detail how attackers used the first attack wave to compromise multiple mailboxes throughout various organizations and implement an inbox rule to evade detection. This was then followed by a second attack wave that abused one organization's lack of MFA protocols to register the attackers' unmanaged device and propagate the malicious messages via lateral, internal, and outbound spam.

First wave and rule creation

The campaign leveraged multiple components and techniques to quietly compromise accounts and propagate the attack. Using Microsoft 365 Defender threat data, we found the attack's initial compromise vector to be a phishing campaign. Our analysis found that the recipients received a DocuSign-branded phishing email, displayed below:

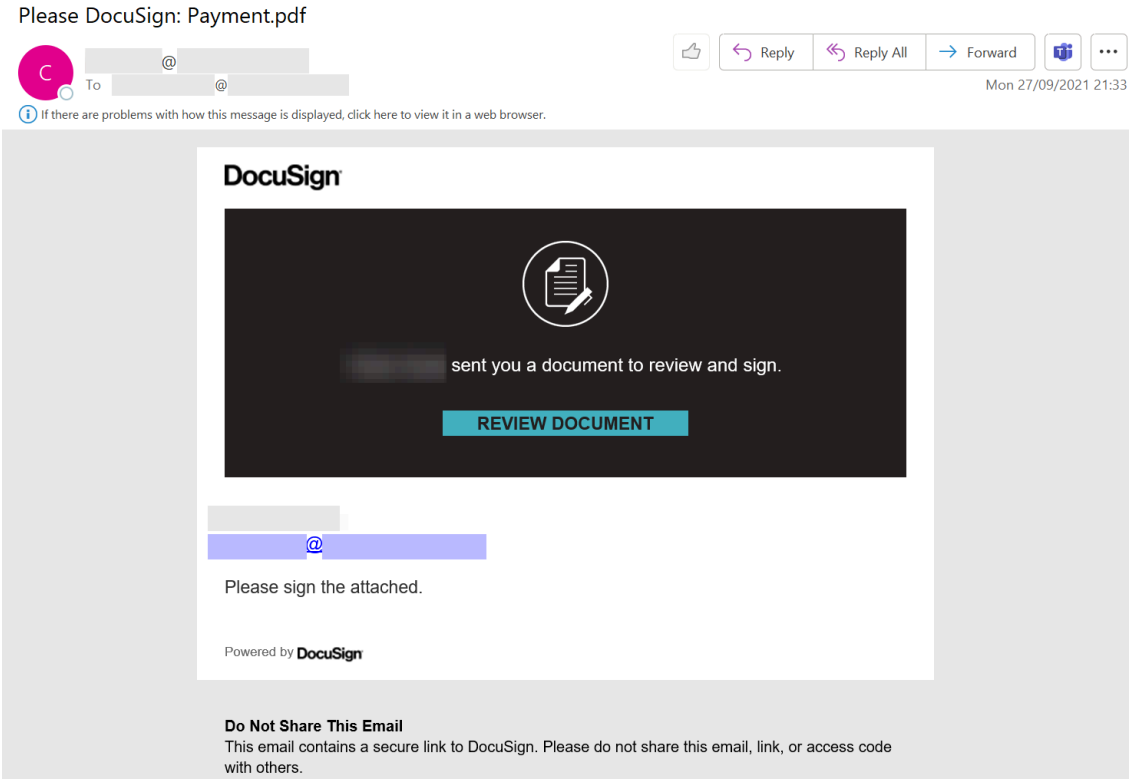

Figure 4. First-stage phishing email spoofing DocuSign

The attacker used a set of phishing domains registered under .xyz top-level domain. The URL domain can be described with the following regular expression syntax:

UrlDomain matches regex @"^[a-z]{5}\.ar[a-z]{4,5}\.xyz"

The phishing link was uniquely generated for each email, with the victim's email address encoded in the query parameter of the URL. After clicking the link, the victim was redirected to a phishing website at newdoc-lnpye[.]ondigitalocean[.]app, which imitated the login page for Office 365. The fake login page was pre-filled with the targeted victim's username and prompted them to enter their password. This technique increased the likelihood that the victim viewed the website as being legitimate and trustworthy.

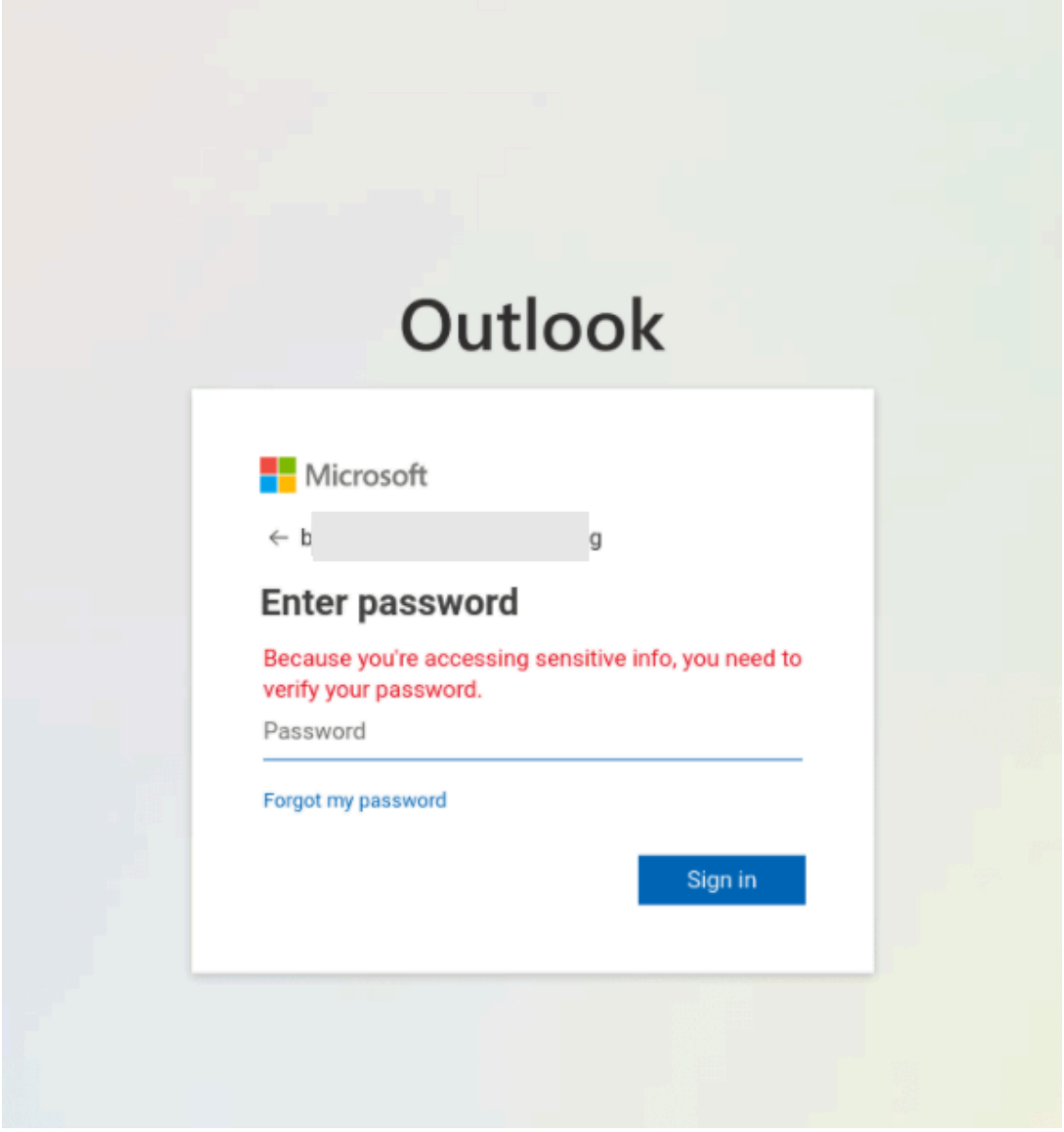

Figure 5. Phishing page with username prepopulated

Next, we detected that the victim's stolen credentials were immediately used to establish a connection with Exchange Online PowerShell, most likely using an automated script as part of a phishing kit. Leveraging the Remote PowerShell connection, the attacker implemented an inbox rule via the *New-InboxRule* cmdlet that deleted certain messages based on keywords in the subject or body of the email message. The inbox rule allowed the attackers to avoid arousing the compromised users' suspicions by deleting non-delivery reports and IT notification emails that might have been sent to the compromised user.

During our investigation of the first stage of this campaign, we saw over one hundred compromised mailboxes in multiple organizations with inbox rules consistently fitting the pattern below:

Mailbox rule name	**Condition**	**Action**

Spam Filter	SubjectOrBodyContainsWords: "junk;spam;phishing;hacked;password;with you"	DeleteMessage, MarkAsRead

While multiple users within various organizations were compromised in the first wave, the attack did not progress past this stage for the majority of targets as they had MFA enabled. The attack's propagation heavily relied on a lack of MFA protocols. Enabling MFA for Office 365 applications or while registering new devices could have disrupted the second stage of the attack chain.

Device registration and second wave phishing

One account belonging to an organization without MFA enabled was further abused to expand the attackers' foothold and propagate the campaign. More specifically, the attack abused the organization's lack of MFA enforcement to join a device to its Azure Active Directory (Azure AD) instance, or possibly to enroll into a management provider like Intune to enforce the organization's policies based on compliant devices.

In this instance, the attackers first installed Outlook onto their own Windows 10 machine. This attacker-owned device was then successfully connected to the victim organization's Azure AD, possibly by simply accepting Outlook's first launch experience prompt to register the device by using the stolen credentials. An Azure AD MFA policy would have halted the attack chain at this stage. Though for the sake of comprehensiveness, it should be noted that some common red team tools, such as *AADInternals* and the command *Join-AADIntDeviceToAzureAD*, can be used to achieve similar results in the presence of a stolen token and lack of strong MFA policies.

Azure AD evaluates and triggers an activity timestamp when a device attempts to authenticate, which can be reviewed to discover freshly registered devices. In our case, this includes a Windows 10 device either Azure AD joined or hybrid Azure AD joined and active on the network. The activity timestamp can be found by either using the *Get-AzureADDevice* cmdlet or the Activity column on the devices page in the Azure portal. Once a timeframe is defined and a potential rogue device is identified, the device can be deleted from Azure AD, preventing access to resources using the device to sign in.

The creation of the inbox rule on the targeted account coupled with the attackers' newly registered device meant that they were now prepared to launch the second wave of the campaign. This second wave appeared to be aimed at compromising additional accounts by sending lateral, internal, and outbound phishing messages.

By using a device now recognized as part of the domain coupled with a mail client configured exactly like any regular user, the attacker gained the ability to send intra-organizational emails that were missing many of the typical suspect identifiers. By removing enough of these suspicious message elements, the attacker thereby significantly expanded the success of the phishing campaign.

To launch the second wave, the attackers leveraged the targeted user's compromised mailbox to send malicious messages to over 8,500 users, both in and outside of the victim organization. The emails used a SharePoint sharing invitation lure as the message body in an attempt to convince recipients that the "Payment.pdf" file being shared was legitimate.

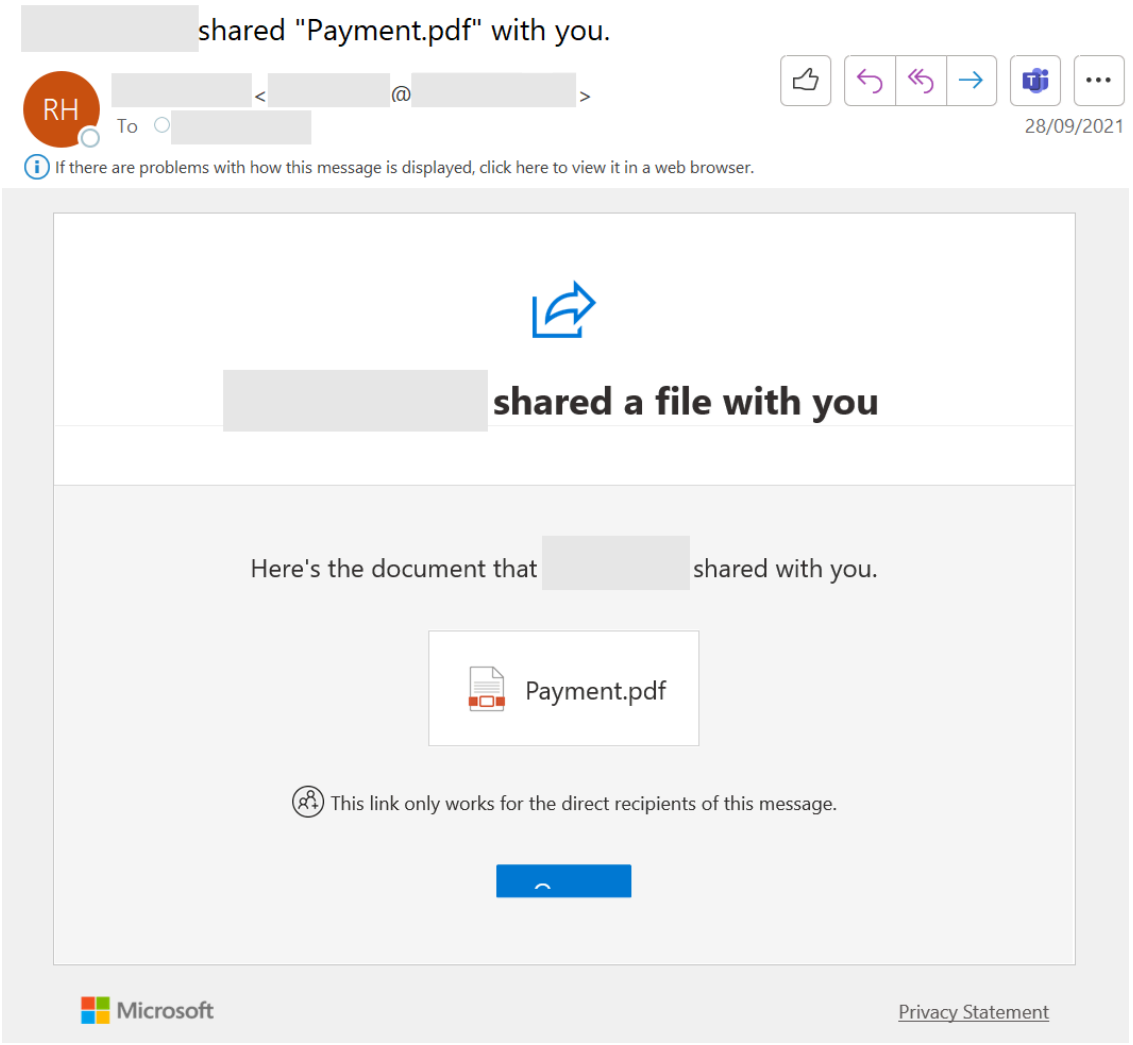

Figure 6. Second-stage phishing email spoofing SharePoint

Like the first stage of the campaign, we found that the URL used in the second wave phishing emails matched the first's wave structure and also redirected to the newdoc-lnpye[.]ondigitalocean[.]app phishing website imitating the Office 365 login page. Victims that entered their credentials on the second stage phishing site were similarly connected with Exchange Online PowerShell, and almost immediately had a rule created to delete emails in their respective inboxes. The rule had identical characteristics to the one created during the campaign's first stage of attack.

Generally, the vast majority of organizations enabled MFA and were protected from the attackers' abilities to propagate the attack and expand their network foothold. Nonetheless, those that do not have MFA enabled could open themselves up to being victimized in potential future attack waves.

Analysis of this novel attack chain and the additional techniques used in this campaign indicates that the traditional phishing remediation playbook will not be sufficient here. Simply resetting compromised accounts' passwords may ensure that the user is no longer compromised, but it will not be enough to eliminate ulterior persistence mechanisms in place.

Careful defenders operating in hybrid networks need to also consider the following steps:

- Revocation of active sessions and any token associated with the compromised accounts
- Deletion of any mailbox rules eventually created by the actor
- Disable and removal of any rogue device joined to Azure AD by the actor

If these additional remediation steps are not taken, the attacker could still have valuable network access even after successfully resetting the password of the compromised account. An in-depth understanding of this attack is necessary to properly mitigate and defend against this new type of threat.

Defending against multi-staged phishing campaigns

The latest Microsoft Digital Defense Report detailed that phishing poses a major threat to both enterprises and individuals, while credential phishing was leveraged in many of the most damaging attacks in the last year. Attackers targeting employee credentials, particularly employees with high privileges, typically use the stolen data to sign into other devices and move laterally inside the network. The phishing campaign we discussed in this blog exemplifies the increasing sophistication of these attacks.

In order to disrupt attackers before they reach their target, good credential hygiene, network segmentation, and similar best practices increase the "cost" to attackers trying to propagate through the network. These best practices can limit an attacker's ability to move laterally and compromise assets after initial intrusion and should be complemented with advanced security solutions that provide visibility across domains and coordinate threat data across protection components.

Organizations can further reduce their attack surface by disabling the use of basic authentication, enabling multi-factor authentication for all users, and requiring multi-factor authentication when joining devices to Azure AD. Microsoft 365 global admins can also disable Exchange Online PowerShell for individual or multiple end users via a list of specific users or filterable attributes, assuming that the target accounts all share a unique filterable attribute such as Title or Department. For additional security, customers can enforce our new Conditional Access control requiring MFA to register devices, which can be combined with other CA conditions like device platform or trusted networks.

Microsoft 365 Defender correlates the alerts and signals related to initial phishing generated by suspicious inbox rule creation as well as suspicious device registration into a single easy to comprehend Incident.

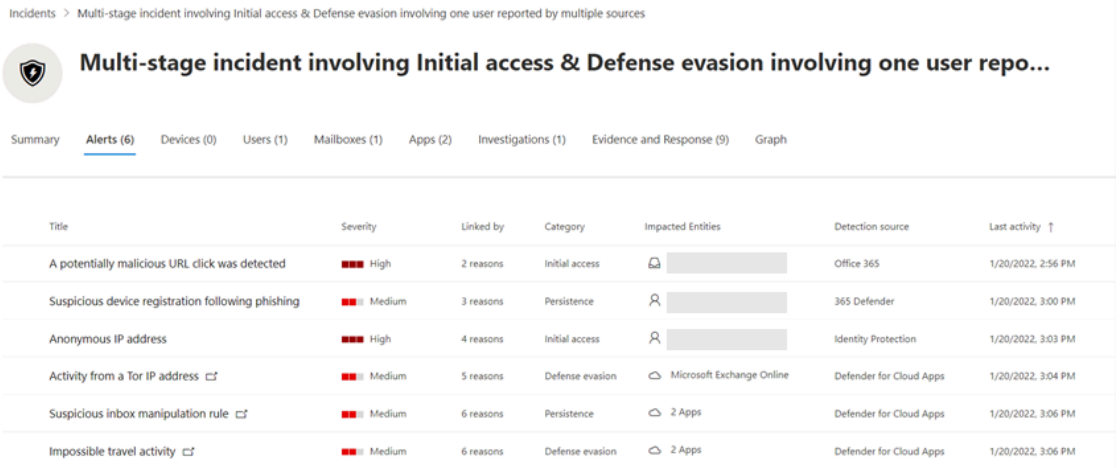

Figure 7. Microsoft 365 Defender incident with suspicious device registration and inbox rule

Microsoft Defender for Office 365 protects against email threats using its multi-layered email filtering stack, which includes edge protection, sender intelligence, content filtering, and post-delivery protection, in addition to including outbound spam filter policies to configure and control automatic email forwarding to external recipients. Moreover, Microsoft Defender for Office 365 uses Safe Links feature to proactively protect users from malicious URLs in internal messages or in an Office document at time of click. Safe Links feature to proactively protect users from malicious URLs in internal messages or in an Office document at time of click.

Advanced hunting queries

Hunting for emails with phishing URL

```
let startTime = ago(7d);
let endTime = now();
EmailUrlInfo
| where Timestamp between (startTime..endTime)
| where UrlDomain matches regex @"^[a-z]{5}\.ar[a-z]{4,5}\.xyz"
| project NetworkMessageId,Url
| join (EmailEvents
| where Timestamp between (startTime..endTime))
on NetworkMessageId
```

Hunting for suspicious Inbox Ruleslet startTime = ago(7d);

```
// Hunting for suspicious Inbox Rules
let startTime = ago(7d);
let endTime = now();
CloudAppEvents
| where Timestamp between(startTime .. endTime)
| where ActionType == "New-InboxRule"
| where RawEventData contains "Spam Filter"
| where RawEventData has_any("junk","spam","phishing","hacked","password","with you")
| where RawEventData contains "DeleteMessage"
```

```
| project Timestamp, AccountDisplayName, AccountObjectId, IPAddress
```

Hunting for rogue device registrations

```
// Hunting for rogue device registrations

let startTime = ago(7d);

let endTime = now();

CloudAppEvents

| where Timestamp between(startTime .. endTime)

| where ActionType == "Add registered owner to device."

| where RawEventData contains "notorius"

| where AccountDisplayName == "Device Registration Service"

| where isnotempty(RawEventData.ObjectId) and isnotempty(RawEventData.ModifiedProperties[0].NewValue)
and isnotempty(RawEventData.Target[1].ID) and isnotempty(RawEventData.ModifiedProperties[1].NewValue)

| extend AccountUpn = tostring(RawEventData.ObjectId)

| extend AccountObjectId = tostring(RawEventData.Target[1].ID)

| extend DeviceObjectId = tostring(RawEventData.ModifiedProperties[0].NewValue)

| extend DeviceDisplayName = tostring(RawEventData.ModifiedProperties[1].NewValue)

| project Timestamp,ReportId,AccountUpn,AccountObjectId,DeviceObjectId,DeviceDisplayName
```
