# CVE-2024-21378 в MS Outlook

*Александр Родченко*

В начале этого года была обнаружена уязвимость СVE-2024-21378, при успешной эксплуатации которой удалённый злоумышленник может выполнить произвольный код в системе, где установлен MS Outlook. Я проанализировал процесс эксплуатации этой уязвимости – и написал утилиту, которая позволяет выявить подозрительную активность, связанную с этой уязвимостью.

## Вид уязвимости

С точки зрения сложности мониторинга (то есть на взгляд аналитика SOC), уязвимости удалённого выполнения кода (RCE) можно разделить на три категории:

1. те RCE, что вызывают старт процесса,
2. те, что загружают библиотеку/скрипт в уязвимый процесс,
3. и те, которые просто тихонько хостят недетектящийся шеллкод в памяти и не отсвечивают лишний раз.

По этой классификации CVE-2024-21378 – уязвимость второго типа. В оригинальном [листе](https://gist.github.com/Homer28/7f3559ff993e2598d0ceefbaece1f97f), кстати, в качестве полезной нагрузки использовалась [библиотека](https://github.com/Homer28/easy_shellcode_generator) с шеллкодом, который только лишь «резолвил» хост вида "new.d%USERDOMAIN%.u%COMPUTERNAME%.attacker.com". Это лишний раз подчёркивает, что эксплоит достаточно беспалевный: нет старта никакого «злого» процесса. А значит, никакие дефолтные правила на старт процесса от Outlook нам не помогут.

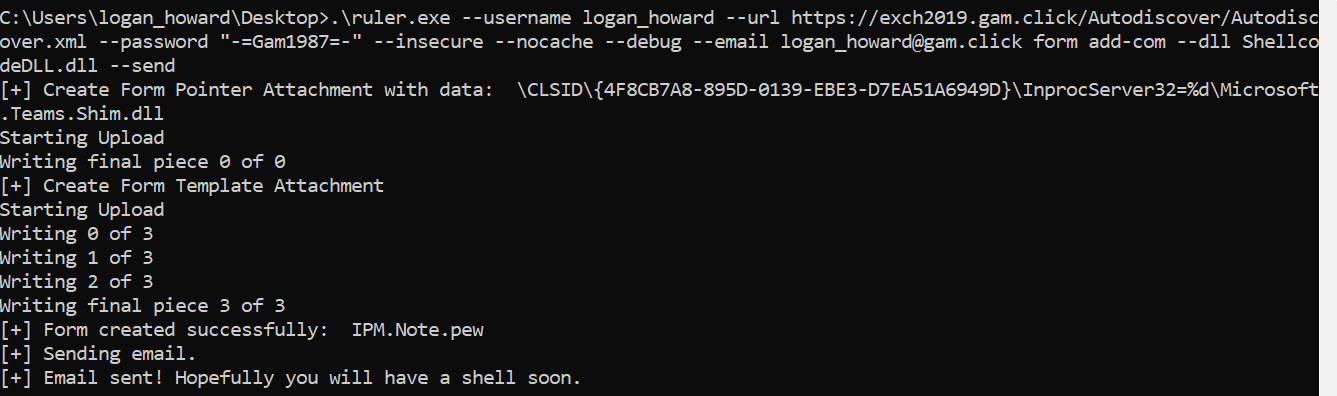
Давайте разберёмся, что же происходит? Если кратко, то Exchange и Outlook общаясь по протоколу MAPI, могут пересылать друг другу не только сообщения ([MessageClass “IPM.Note](https://learn.microsoft.com/en-us/office/vba/outlook/concepts/forms/form-name-and-message-class-overview) ”), но и контакты, приглашения на встречи, задачи, логи, отчёты и [многое другое](https://learn.microsoft.com/en-us/office/vba/outlook/concepts/forms/item-types-and-message-classes), зачастую вообще [немыслимое](https://learn.microsoft.com/en-us/office/vba/outlook/concepts/forms/form-name-and-message-class-overview).

В частности, существует возможность создать свой собственный тип сообщения и указать клиенту Outlook, что для его отрисовки следует использовать какую-то особую форму. Эту форму также можно передать в виде сообщения. Такому сообщению соответствует класс [IPM.Microsoft.FolderDesign.FormsDescription](https://github.com/NetSPI/ruler/blob/44d1b60a343054e11607db7c6485dd02774a729d/forms/rulerforms.go#L121), а сама форма представляет собой DLL-библиотеку, которая хранится как вложение в этом сообщении.

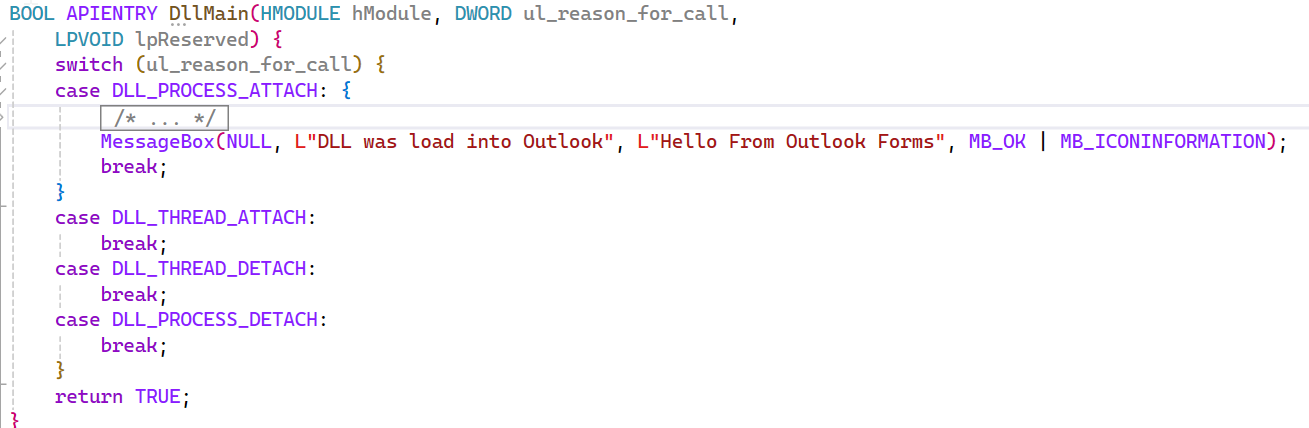
Это даже сложно назвать уязвимостью – по всей видимости, именно так и планировалось для удобства. Другое дело, что в этой библиотеке может быть вредоносный код, который Outlook выполнит без всяких проверок.

## Как проводится атака

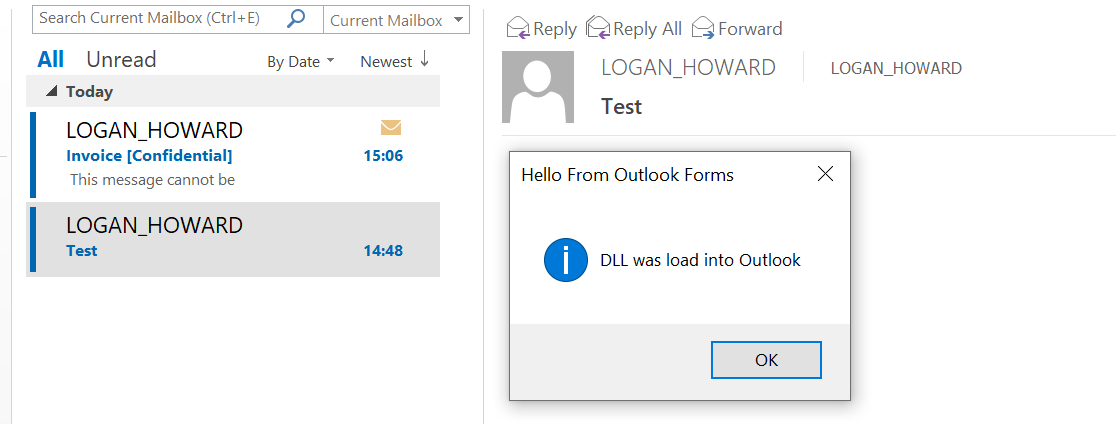
Чтобы проэксплуатировать данную уязвимость, злоумышленник создаёт кастомную форму (библиотека Shellcode.DLL) и посылает её жертве:



В нашем тестовом примере ShellcodeDLL не делает ничего особенного – просто выводит сообщение в Message Box:

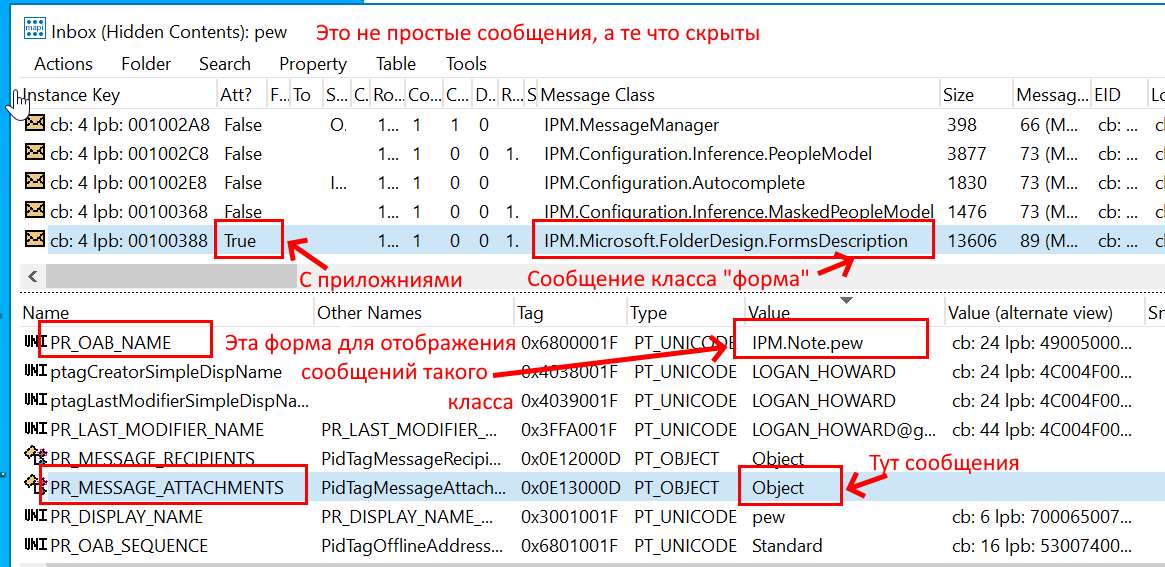


После отправки формы, которая открывает определённый класс сообщений, злоумышленник посылает жертве письмо-триггер, принадлежащее к заданному классу сообщений. В результате при открытии Outlook видим выполнение отправленной библиотеки:

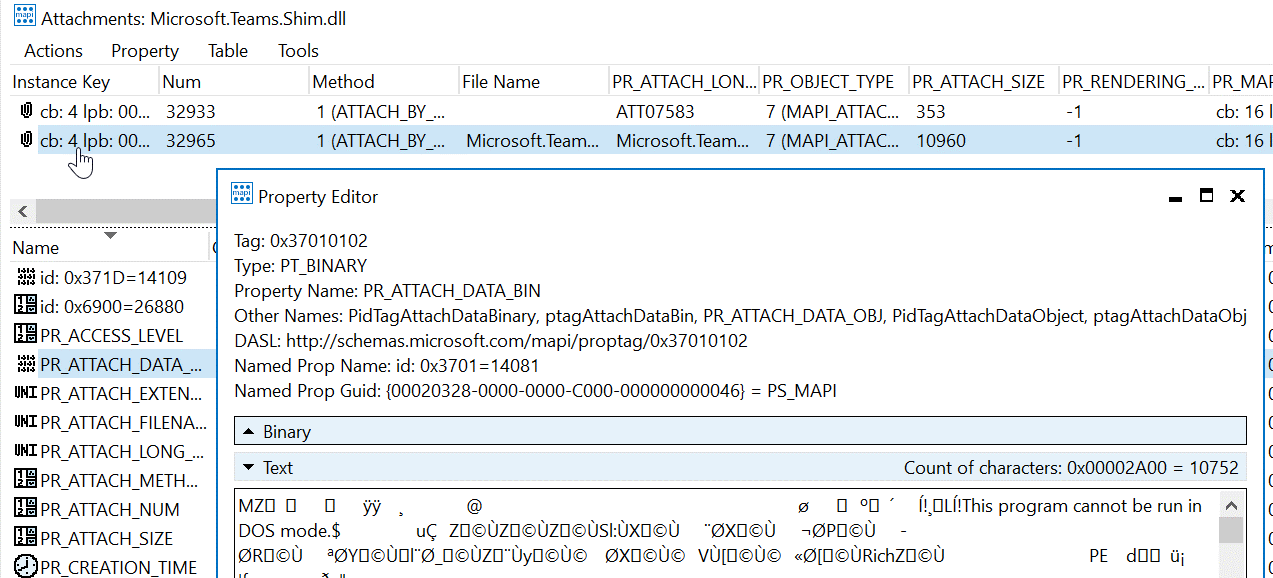


## Как найти вредоносную форму

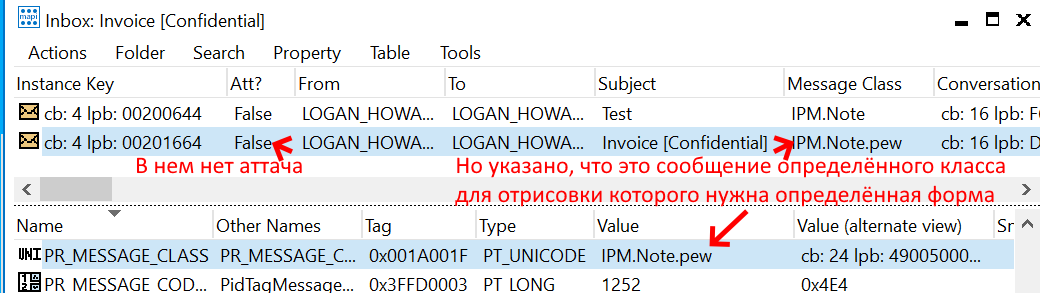
Как уже было сказано, письмо с дефолтной темой «Invoice [Confidential]» не содержит в себе ничего злодейского. Это лишь триггер, который загрузит форму. А где же сама форма? Воспользуемся MFCMAPI и посмотрим на сообщение с формой:



Можно увидеть, что во вложении к письму содержится библиотека:

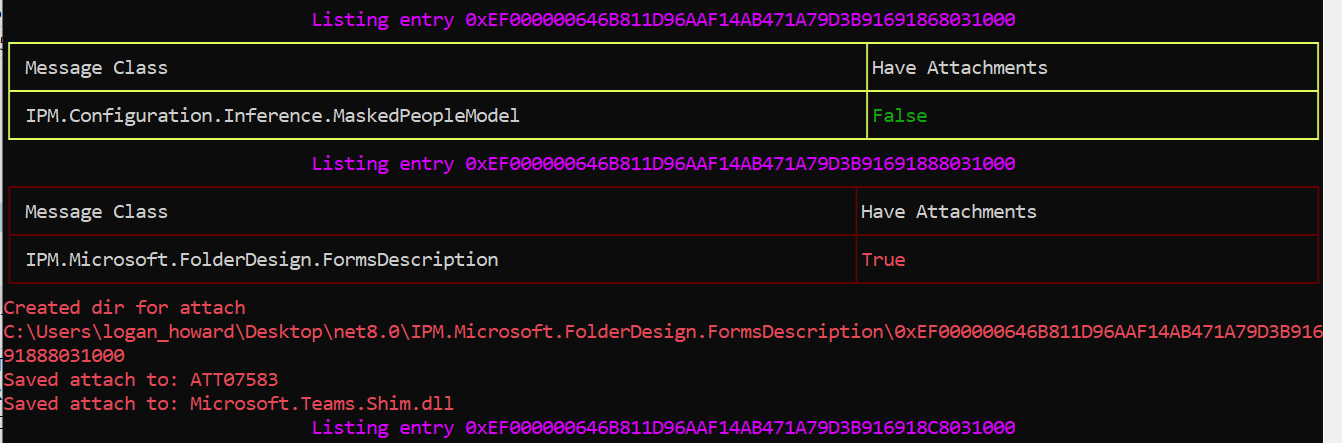


Теперь посмотрим на письмо с триггером:



## Детектирование попыток эксплуатации

Я написал [утилиту](https://github.com/gam4er/OutlookFormFinder), которая сканирует скрытые сообщения (это различные служебные сообщения, включая правила обработки писем в папке "Входящие") и при обнаружении в них вложения сохраняет его в текущую директорию (CWD):



Компания Microsoft устранила уязвимость СVE-2024-21378 с помощью обновления безопасности, выпущенного в феврале 2024 года. Для снижения риска необходимо установить последние обновления для Microsoft Outlook. В частности, исправление включено в различные сборки Outlook и Office, причем Outlook 2016 получил обновление в мае 2024 года (смотрите детали в [CVE CyberSecurity Database News](https://www.cve.news/cve-2024-21378/), [Microsoft Support](https://support.microsoft.com/en-us/office/recall-message-in-outlook-desktop-stops-working-after-february-2024-security-updates-170ae542-c5b8-4681-a8f1-b44895e7ef12), [NetSPI](https://www.netspi.com/blog/technical-blog/adversary-simulation/microsoft-outlook-remote-code-execution-cve-2024-21378/) и [BornCity](https://borncity.com/win/2024/03/12/microsoft-outlook-rce-vulnerability-cve-2024-21378-patched-in-february-2024/)).

Однако, даже если ваша версия Outlook обновлена и неуязвима для эксплуатации CVE-2024-21378, вредоносные формы могут быть уже доставлены в вашу почтовую систему до установки исправления (и даже после, просто эксплуатации не будет). Поэтому важно знать, как обнаружить и удалить такие формы и сомнительные вложения. Для этих целей и пригодится [моя утилита](https://github.com/gam4er/OutlookFormFinder), которая “ретроспективно” просмотрит все скрытые сообщения и извлечёт вложения свет божий для анализа. Если вы увидите в качестве вложения исполняемый файл, скрипт или библиотеку – это повод как минимум изучить их.