August 2022
Vol. XLI, No 08

KEYSTONE MacCentral

printout

Keystone MacCentral Macintosh Users Group ❖ www.keystonemac.com

August Meeting

Please see your membership email for the links to this month's Zoom meeting or email us at KeystoneMacCentral@mac.com.

During our program this month we plan to discuss

- Migrating new data to a Mac
especially for those investing a new Mac

- Top 40 keyboard shortcuts
Maybe you can find some new & useful shortcuts

- Blink Home Security System

Blink is an ecosystem of economical devices (both indoor & outdoor) that watches over your home and communicates with you via an app.

We have virtual meetings via Zoom
on the third Tuesday of each month.

Emails will be sent out prior to each meeting.
Follow the directions/invitation each month
on our email – that is, just click on the link
to join our meeting.

Contents

By Glenn Fleishman

Why Passkeys Will Be Simpler and More Secure Than Passwords

Apple has unveiled its version of passkeys, an industry-standard replacement for passwords that offers more security and protection against hijacking while simultaneously being far simpler in nearly every respect.

You never type or manage the contents of a passkey, which is generated when you upgrade a particular website account from a password-only or password and two-factor authentication login. Passkeys overcome numerous notable weaknesses with passwords:

- Each passkey is unique—always.
- Every passkey is generated on your device, and the secret portion of it never leaves your device during a login. (You can securely sync your passkeys across devices or share them with others.)
- Because passkeys are created using a strong encryption algorithm, you don't have to worry about a "weak" password that could be guessed or cracked.
- A website can't leak your authentication credentials because sites store only the public component of the passkey that corresponds to your login, not the secret part that lets you validate your identity.
- An attacker can't phish a passkey from you because a passkey only presents itself at a legitimately associated website.
- Passkeys never need to change because they can't be stolen.
- Passkeys don't require two-factor authentication because they incorporate two different factors as part of their nature.

After a test run with developers over the last year, Apple has built passkey support into iOS 16, iPadOS 16, macOS 13 Ventura, and watchOS 9, slated for release in September or October of this year. These operating systems will store passkeys just as they do passwords and other entries in the user keychain, protected by a device password or passcode, Touch ID, or Face ID. Passkeys will also sync securely among your devices using iCloud Keychain, which employs end-to-end encryption—Apple never has access to passkeys or other iCloud Keychain data.

Best of all, perhaps, is that Apple built passkeys on top of a broadly supported industry standard, the W3C Web Authentication API or WebAuthn, created by the World Wide Web Consortium and the FIDO Alliance, a group that has spent years developing approaches to reduce the effectiveness of phishing, eliminate hijacking, and increase authentication simplicity for users. Apple, Amazon, Google, Meta (Facebook), and Microsoft are all FIDO board members, as are major financial institutions, credit card networks, and chip and hardware firms.

Many websites and operating systems already support WebAuthn via a hardware key like the popular ones made by Yubico. You visit a website, choose to log in using a security key, insert or tap a button on the hardware key, and the browser, operating system, and hardware key all talk together to complete the login. A passkey migrates the function of that hardware key directly into the operating system—no extra hardware required. Websites that already support hardware-based WebAuthn should be able to support passkeys with little to no effort, according to Apple.

Before we get started, note that Apple writes "passkey" in lowercase, an attempt to get us to use it alongside password, passcode, and passphrase as a common concept. Google, Microsoft, and other companies will offer compatible technology and may also opt for the generic passkey name. While

new terminology can cause confusion, "passkey" is better than the more technically descriptive "multi-device FIDO credentials," which doesn't exactly roll off the tongue.

Let's dig in to how passkeys work.

Passkeys Bring the Benefits of Public-Key Cryptography to Everyday Logins

Passkeys rely on public-key cryptography, something we've been writing about at TidBITS for nearly 30 years. With public-key cryptography, an encryption algorithm generates a secret that's broken into two pieces: a private key, which you must never disclose, and a public key, which you can share in any fashion without risk of exposing the private key. Public-key cryptography underpins secure Web, email, and terminal connections; iMessage; and many other standards and services.

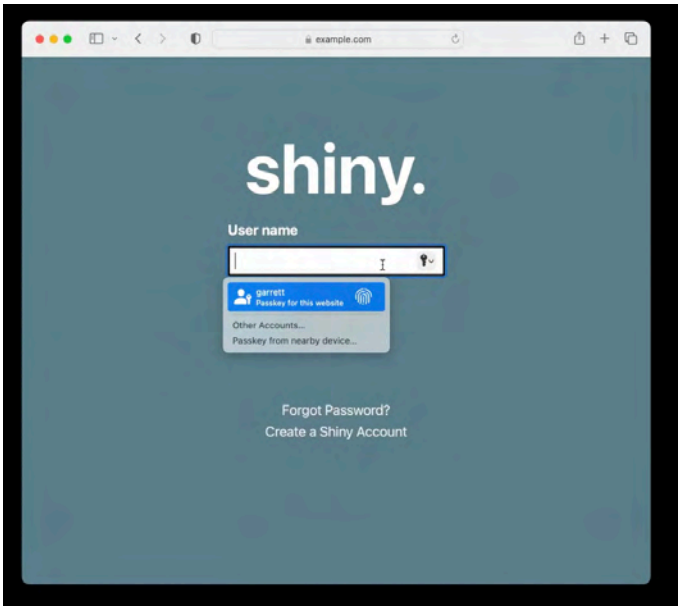
Anyone with a person's public key can use it to encrypt a message that only the party who possesses the private key can decrypt. The party who has the private key can also perform a complementary operation: they can "sign" a message with the private key that effectively states, "I validate that I sent this message." Crucially, anyone with the public key can confirm that *only* the private key's possessor could have created that signature.

A passkey is a public/private key pair associated with some metadata, such as the website domain for which it was created. With a passkey, the private key never leaves the device on which it was generated to validate a login, while a website holds only the corresponding public key, stored as part of the user's account.

To use a passkey, the first step is to enroll at a website or in an app. You're likely familiar with this process from any time you signed up for two-factor authentication at a site: you log in with existing credentials, enable 2FA, receive a text message or scan a QR code into an authentication app or your keychain (in iOS 15, iPadOS 15, and Safari 15 for macOS), and then verify your receipt.

With a passkey, the process is different. When you log in to a website offering passkey authentication, you will have an option to upgrade it to a passkey in your account's security or password section. The website first generates a registration message that Apple's operating systems will interpret—it happens at a layer you never see. In response, your device creates the public/private key pair, stores it securely and locally, and transmits the public key to the website. The site can then optionally issue a challenge for it and your device can present it to confirm the enrollment.

On subsequent visits, when you're presented with a login, your iPhone or iPad will show the passkey entry in the QuickType bar and Safari in macOS will show it as a pop-up menu. In both cases, that's just like passwords and verification codes today. As with those login aids, you'll validate the use of your passkey with Touch ID, Face ID, or your device passcode, depending on your settings.



Source: Apple

Behind the scenes, your request to login via a passkey causes the site server to generate a "challenge" request using the stored public key. Your device then has to build a response using your stored private key. Because you initiate a passkey login by validating your identity, your device has access to your passkey's private key when the

challenge request comes in and can respond to the challenge without another authentication step. The server validates your device's response against your stored public key, ensuring that you are authorized for access. If it all checks out, the website logs you in.

A passkey replaces two-factor authentication, and it's worth breaking down why, as it seems counter-intuitive: how can a single code held on a device provide distinct aspects of confirmation? The rubric for multiple security factors is usually stated as at least two of "something you know, something you have, or something you are." A passkey incorporates at least two of those:

- **Something You Know:** While commonly thought of as a password, the "know" part is really any fixed piece of information you possess. Think of a 20-character randomly generated password stored in your password manager. Do you "know" that? Yes, in the sense that it's retrievable exactly as entered.
- **Something You Have:** Because passkeys are locked to devices, you prove your possession of a device by unlocking the passkey: no device, no passkey.
- **Something You Are:** Although passkeys don't require biometric authentication using Face ID or Touch ID, it's an option. Apple always lets you use a device passcode to backstop Face ID or Touch ID, so it's a blurred line with "something you know" compared to a dedicated biometric device with no fallback option.

Think for a moment about the advantages here. A passkey:

- **Resists phishing:** As with passwords and verification codes, your device will only present a passkey in QuickType or as a pop-up menu option when on a website's specific domain associated with the passkey. An attacker can't fool you into entering a passkey on a deceptive site, as can be done with a password.
- **Prevents reuse of a stolen key on other accounts:** Because each passkey is unique to its associated website, even if the site suffers a security breach, the only credential that can be stolen is your public key. That public key is useless to help a thief log in as you at another site as they lack your private key to answer the login challenge.
- **Blocks damage from malicious code injection on a website you visit:** A malicious party can often "inject" malicious JavaScript onto an otherwise benign page. It has happened at times even to major websites, usually due to poorly vetted malicious ads delivered automatically through self-service advertising networks. A website that falls prey to just a front-end attack on its HTML and scripts wouldn't allow the attacker to produce a valid challenge request for your device's passkey. The site would also have to suffer from a back-end compromise of its server code for account information to be at risk, at which point the site's data is probably fully compromised anyway.
- **Blocks guessing, identity searching, brute force:** Because every passkey has a super-complex secret, an attacker can't successfully guess or brute force your access to a site.
- **Eliminates 2FA hijacking:** Because passkeys don't have a second factor, they aren't vulnerable to SMS hijacking and interception, site impersonation and phishing, and other techniques to acquire a second factor.

Apple stores each passkey as just another entry in your keychain. If you have iCloud Keychain enabled, the passkeys sync across all your devices. (iCloud Keychain requires two-factor authentication enabled on your Apple ID; Apple hasn't said if passkeys will replace its internal use of 2FA for its user accounts.)

You can share a passkey with someone else using AirDrop. This means you have to be in proximity to the other person, another element in security. The details are shared through end-to-end encryption, allowing the private key and other data to be passed without risk of interception. Apple hasn't provided much more detail than that AirDrop

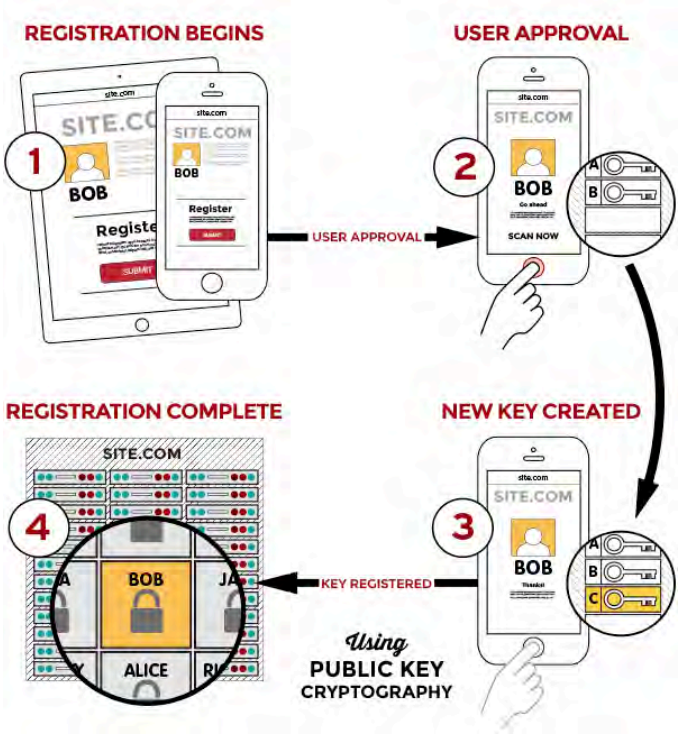
sharing is an option, so there may be other provisos or security layers.

Because passkeys replace passwords and a second factor, you may be reasonably worried at this point about losing access to your passkeys if you're locked out of your Apple ID account or lose all your registered devices. Apple has several processes in place for recovering Apple ID account access and broad swaths of iCloud-synced data. For an Apple ID account, you can use Apple's account recovery process or an account Recovery Key. For iCloud data, if you've enabled the friends-and-family recovery system, iCloud Data Recovery Service, you can use that to re-enable access. After you recover account access, Apple has an additional set of steps that enable you to retrieve iCloud Keychain entries: it involves sending a code via SMS to a registered phone number and entering a device passcode for one of the devices in your iCloud-synced set.

This is all a fabulous reduction in the potential for successful attacks against your Internet-accessible accounts. But there's more: Apple isn't building yet another walled garden. Instead, passkeys are part of a broad industry effort with which Apple says its implementation will be compatible.

Passkeys Are an Industry Standard, Not a Proprietary Technology

Apple built its passkey support on top of the previously mentioned WebAuthn standard, which describes the server side of how to implement a Web-based login with public-key cryptography. FIDO created standards for the client side of that equation and calls the combination of its protocol and WebAuthn FIDO2. Apple developed its own client-side approach that's compatible with standard WebAuthn servers and should be interchangeable with other companies' rollouts of passkeys. Google, Microsoft, and Apple made a joint announcement in May 2022 committing to this approach, too.



FIDO's schematic for a generalized registration process. (Source: FIDO Alliance)

In Apple's passkey introduction video for developers, engineer Garrett Davidson emphasized Apple's commitment to compatibility, saying:

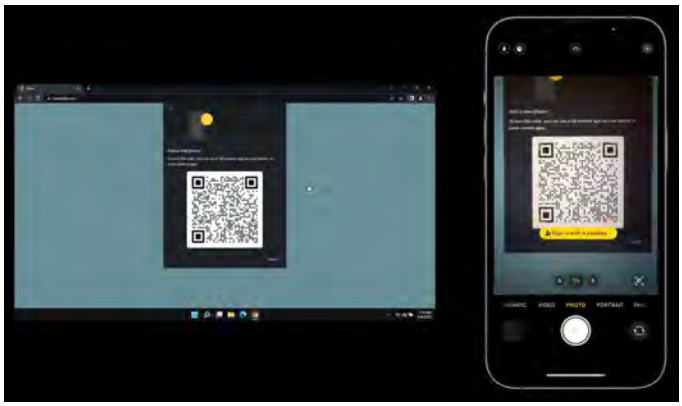
We've been working with other platform vendors within the FIDO Alliance to make sure that passkey implementations are compatible cross-platform and can work on as many devices as possible.

He then demonstrated using a passkey on an Apple device to log in to a website on a PC, showing how a QR code could be used to enable a passkey login to one of your accounts on a device or browser that's not connected to your existing devices or ecosystem.

Here's how you might log in to a passkey-enabled account on someone else's PC using your iPhone with your passkey as the authenticator. During the login, you can opt to add a device instead of entering a passkey or other authentication in the browser. The website's server generates a QR code that includes a pair of single-use passwords—they're generated just for that login and used in the next step for additional validation. (Note that the

device with the browser could be any passkey-supporting operating system and device. The authenticating devices might be limited by Apple or other companies to a smaller set, much like you can only use an iPhone to confirm Apple Pay in Safari on a Mac, not a Mac with Touch ID to confirm Apple Pay from an iPhone.)

The PC in our example also starts broadcasting a Bluetooth message that contains the information needed to connect and authenticate directly with the server. Scan that QR code on your iPhone, and the iPhone uses an end-to-end encrypted protocol to create a tunnel with the PC's Web browser using the keys shown in the QR code. (This encrypted connection isn't part of the Bluetooth protocol, by the way, but data tunneled over Bluetooth; Bluetooth doesn't incorporate the necessary encryption strength.)



Source: Apple

This Bluetooth connection provides additional security and verification by offering *out-of-band* elements, or details that the PC isn't presenting to the device that's providing authentication—here, your iPhone. Because Web pages can be spoofed for phishing attacks, the Bluetooth connection provides a device-to-device backchannel for key details:

- **Server addresses:** The QR code doesn't tell the iPhone what server (or list of servers) it can connect to for the actual passkey connection. That prevents a browser from providing malicious information.
- **Key validation:** The successful creation of an end-to-end encrypted two-way session over Bluetooth using the keys in the QR code enables the iPhone and PC to confirm that the QR code the browser delivered and the iPhone scanned are identical. (Apple hasn't yet provided full details on this stage. The operating system clearly generates the QR code based on a request from the browser, and the browser can't sniff the Bluetooth connection. So a front-end attack that displayed a malicious QR code wouldn't work, as the PC and iPhone communicate without the browser in the loop.)
- **Proximity:** Connecting over short-range Bluetooth demonstrates, with confidence, that the PC and iPhone are near each other.

This broad device and platform compatibility lets you maintain the same degree of passkey security and simplicity without downgrading to a weaker method for login when accessing your account using other people's devices. Whenever there's a way to force a weaker login method, malicious parties will exploit that via phishing, social engineering, or other interception techniques. (Providing a second factor via an SMS text message versus a verification code is a prime example of a weaker backup approach that has been exploited.) In fact, until passkeys can be used exclusively, password-based logins will have to remain available, and they'll remain vulnerable.

There might be some usability hiccups as passkeys roll out, but they shouldn't be widespread. It's possible, for instance, that some WebAuthn server components will need to be updated or that Apple will have to add more edge cases to its framework to encompass how things work in the wild.

But imagine a world in which you can securely log in to websites using any current browser on any device running any modern operating system, without having to create, remember, type, and protect passwords. It's relaxing just to think about.

The main question that remains unanswered is how portable passkeys will be among ecosystems: can I use iOS and Android and Windows and share a passkey generated on one among all three? Given that Apple has built an AirDrop-sharing method for passkeys, I hope FIDO's broad compatibility

includes sharing passkeys among operating systems, too.

A Return to Security through Proximity

Passwords have provided an uneasy security compromise since their introduction decades ago when multi-user computing systems began to require protection. Passwords are patently imperfect, a relic of an age when physical proximity provided the first level of protection, something rendered moot by the Internet.

In an effort to answer some of the weaknesses in a password system, two-factor authentication was grafted on to require that you had something besides a password, something that required holding or being near an object to validate your right to log into a computer, service, or website. But because 2FA starts with an account password and uses a second method that can be subject to compromise or phishing, it remains a patch applied to a damaged wall.

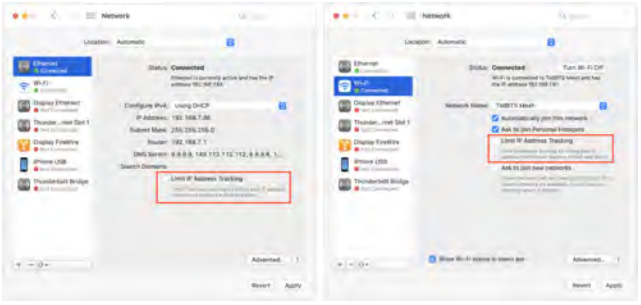
The passkey is a modern replacement for passwords that rebuilds the security wall protecting standard account logins. Proximity—in the form of the device that stores your passkeys—is a powerful tool in reducing account hijacking and interception. Passkeys may seem scary and revolutionary, but they're actually safer and, in some ways, a bit old-fashioned: they're a bit of a throwback to a time when having access to a terminal provided proof you were authorized to use it.

By Adam Engst

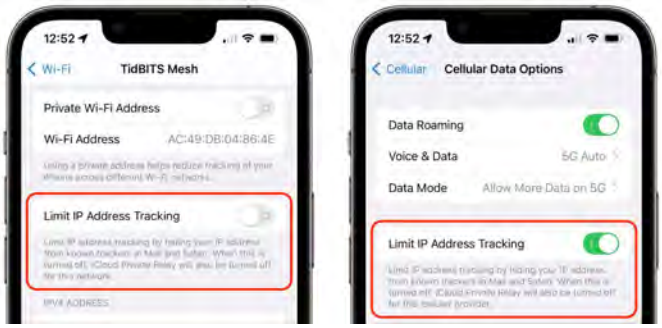
Solving Connectivity Problems Caused by Interlocking Apple Privacy Settings

Complaints about website loading have been trickling in of late, and while the details vary, the commonality has been that the problems started with macOS 12.4 Monterey. Sometimes the problem was just with Safari; other times, it affected Chrome and other browsers too. In some cases, the entire page would refuse to load; in others, only portions of the page would fail.

The solution to the problems I've seen so far is simple: in System Preferences > Network, turn off Limit IP Address Tracking for each network adapter you use (Ethernet and Wi-Fi below—they look surprisingly different).



For some people, the problems have extended to iOS 15 and iPadOS 15. Apple provides the same Limit IP Address Tracking option in Settings > Wi-Fi > *YourNetwork* and Settings > Cellular > Cellular Data Options.



If you read the fine print underneath the iPhone screenshots above, you'll notice that it says, "When this is turned off, iCloud Private Relay will also be turned off for this network." That message appears on my iPhone because I do have iCloud Private Relay enabled for the iPhone, whereas I turned it off on my Mac.

I wish I better understood what's happening here, but it's devilishly difficult to test a feature that prevents tracking by malicious actors, given that I'm neither malicious nor an actor. Clouding the situation even further is the fact that features that say they'll limit IP address tracking or hide your IP address exist in three completely separate places:

- **iCloud Private Relay:** This overarching privacy feature routes all your traffic through two separate Internet relays to hide your IP address from the site to which you're connecting. You can turn it on and off in System Preferences > Apple ID on the Mac and Settings > *YourName* in iOS and iPadOS.
- **Limit IP Address Tracking:** This option is either enabled or disabled for each network you use, whether Wi-Fi, Ethernet, or cellular. As noted above, its description changes depending on whether iCloud Privacy Relay is on or off.
- **Hide IP Address:** Safari and Mail both offer this option in their preferences but say little about how it relates to iCloud Private Relay.

Here's what I think is going on and where I'm unsure. I hope you can use this information to walk the fine line between increased privacy and more frequent connection problems.

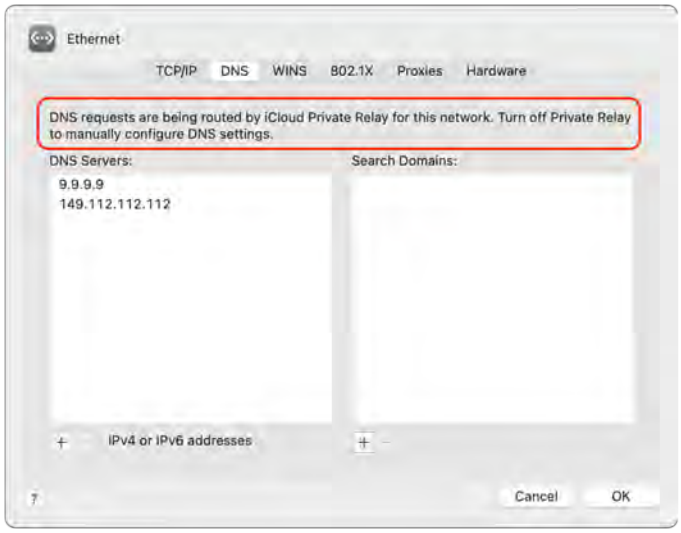
iCloud Private Relay

The first thing to check if you experience sporadic networking failures is iCloud Private Relay. This feature, available only to iCloud+ subscribers who pay Apple for additional storage space, routes all your traffic through two Internet relays, one run by Apple and another run by a major content-delivery network like Akamai, Cloudflare, or Fastly. Apple has a white paper that explains it in detail, but here are the basics.

The privacy win is that only your ISP and Apple know your IP address because your ISP and the first relay (called the "ingress proxy") have to associate the connection with you to send the response back to you. The address of the website you want to load is encrypted, however, so neither your ISP nor Apple knows where you're going.

The second relay (known as the "egress proxy") assigns a new, temporary IP address to the request, decrypts the address of the destination website, and completes the connection to the remote site. In other words, the egress proxy doesn't know your IP address—it gets only enough information to locate you in roughly the right region of the world so geolocation isn't a problem.

Apple acknowledges that iCloud Private Relay can cause problems, in part due to the new transport protocols it uses. iCloud Private Relay also takes over from your DNS servers, which may account for some of the problems; at least one user had a Pi-hole ad blocker installed. macOS tells you this when you specify DNS servers in System Preferences > Network > *YourNetwork* > Advanced > DNS.



As a user, however, if you have problems, there are only two things you need to try, as described above:

- **Disable iCloud Private Relay entirely.** It's easily turned on and off, so there's no harm in flipping that switch as needed.
- **Disable Limit IP Address Tracking for a particular network.** That would let you, for instance, disable it on your iPhone for your home Wi-Fi network while leaving it on for your cellular data connection.

You wouldn't necessarily guess that Limit IP Address tracking would disable iCloud Private Relay for a particular network, and Apple mentions it only once in its documentation of iCloud Private Relay, saying:

*Private Relay can be turned on or off just for a specific network using the Limit IP Address Tracking preference.**

The asterisk points to a footnote that says:

** In earlier versions of iOS, iPadOS, and macOS, this preference is called iCloud Private Relay.*

So why did Apple rename that option? Here's where things get murky. I think it has to do with Limit IP Address Tracking doing more than just disabling iCloud Private Relay.

Limit IP Address Tracking

Apple has said that disabling Limit IP Address Tracking turns iCloud Private Relay off for a particular network. And I think it's safe to say that if you disable both iCloud Private Relay and Limit IP Address Tracking, traffic will flow normally to and from your ISP and destination sites.

But what about the remaining possibility, where iCloud Private Relay is turned off, but Limit IP Address Tracking is turned on? Here's where that fine print comes into play. When iCloud Private Relay is turned on, the fine print reads:

Limit IP address tracking by hiding your IP address from known trackers in Mail and Safari. When this is turned off, iCloud Private Relay will also be turned off for this network.

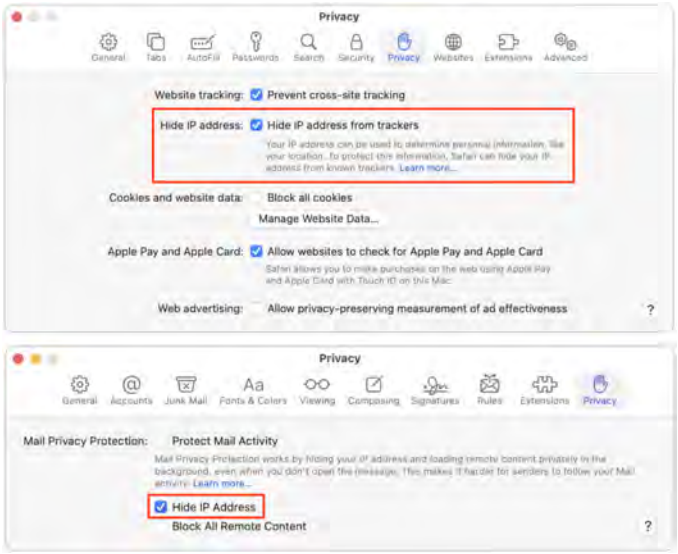
With iCloud Private Relay turned off, the fine print shrinks to:

Limit IP address tracking by hiding your IP address from known trackers in Mail and Safari.

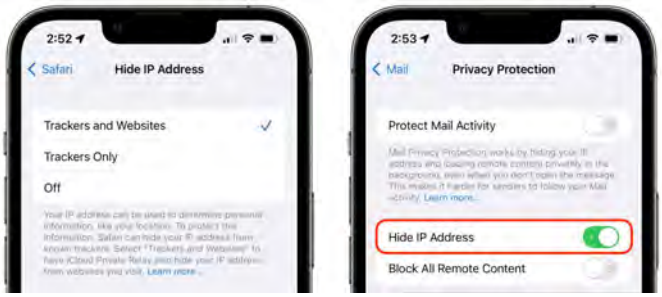
I haven't been able to find any Apple documentation of what this means, but my guess is that Apple has essentially embedded the iCloud Private Relay approach of routing traffic through two Internet relays into Mail and Safari, such that it affects only requests from those apps. What I don't understand is what "hiding your IP address from known trackers" means and how it differs from hiding your IP address in general. Let's investigate.

Hide IP Address

On the Mac, you can go to Safari > Preferences > Privacy to find another Hide IP Address setting. In Mail, look in Mail > Preferences > Privacy, though you must disable Protect Mail Activity to manage the Hide IP Address option separately. (Generally speaking, leave Protect Mail Activity enabled if you can.)



In iOS and iPadOS, you'll find the equivalent options in Settings > Safari > Hide IP Address and Settings > Mail > Privacy Protection. In Mail, again, you must turn off Protect Mail Activity if you want to control Hide IP Address on its own.



So what do these Hide IP Address features do? With Safari, it's difficult to know. If you click or tap the Learn More link on either the Mac or iPhone, it takes you to an explanatory page about iCloud Private Relay that offers no insight into the link to Safari.

Mail, however, is more forthcoming. Click or tap its Learn More link, and you'll get quite a bit of information about how Protect Mail Activity uses a two-hop system that sounds nearly identical to iCloud Private Relay. It even clarifies that if you turn off Protect Mail Activity and leave Hide IP Address enabled, it will continue to "mask your IP address using the same two-separate-internet-relays design."

In addition, Protect Mail Activity routes all remote content downloaded by Mail through two separate relays operated by different entities. The first knows your IP address, but not the remote Mail content you receive. The second knows the remote Mail content you receive, but not your IP address, instead providing a generalized identity to the destination. This way, no single entity has the information to identify both you and the remote Mail content you receive. Senders can't use your IP address as a unique identifier to connect your activity across websites or apps to build a profile about you. … If you choose to disable Protect Mail Activity, the Hide IP Address feature will still mask your IP address using the same two-separate-internet-relays design.

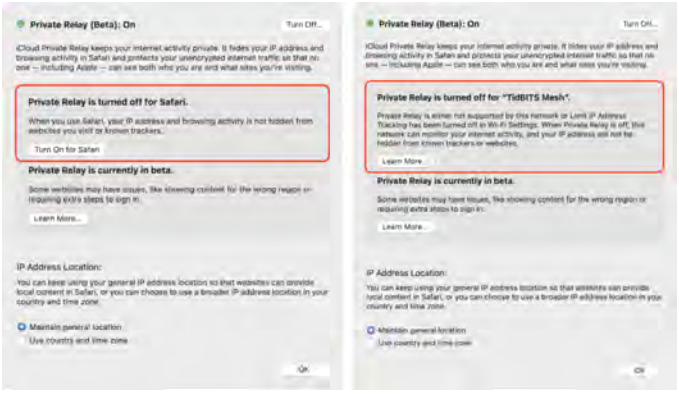
I suspect that it's iCloud Private Relay all the way down.

Putting It All Together

Here's how I believe we should think about these interlocking settings.

- **iCloud Private Relay:** At the top level is iCloud Private Relay. Turn it on, and it runs all your traffic through the ingress and egress proxies, providing the highest level of privacy. However, it's entirely likely that iCloud Private Relay will cause problems, so Apple lets users drop down to a lower level of privacy.
- **Limit IP Address Tracking:** That's where the Limit IP Address Tracking option at the network level comes in. You can disable it to turn off iCloud Private Relay selectively or enable it (with iCloud Private Relay disabled) to apply iCloud Private Relay-like traffic routing to traffic from Safari and Mail. But since those apps are quite different—Safari needs to be able to connect to a far more varied set of servers than Mail—Apple separated them as well.
- **Hide IP Address:** That's why each app has its own Hide IP Address setting. You might need to turn off iCloud Private Relay, turn off Limit IP Address Tracking, and turn off Safari's Hide IP Address setting but still want to keep Mail's Hide IP Address option enabled. It's conceivable you'd want to disable Mail's tracking protection and enable Safari's, but that seems less likely.

Lending support to my theory is that if you disable Hide IP Address for Safari and Limit IP Address Tracking for your network and then turn on iCloud Private Relay, it first prompts you to turn on Safari's Hide IP Address setting (below left) and then alerts you that it's disabled for your network (below right).



Again, I don't know what Apple means when it specifies that Limit IP Address Tracking and Hide IP Address affect only "known trackers." The Hide IP Address screen in Safari makes the distinction clear—as long as iCloud Private Relay is enabled, you can choose from either Trackers and Websites or just Trackers. Without iCloud Private Relay turned on, you can only choose to hide your IP address from Trackers.

I've been unable to find any Apple documentation of how the company identifies known trackers and massages Safari and Mail traffic to protect your IP address from them. What happens when you connect to a remote site that's not a known tracker? Does Apple send your IP address through in the clear? Perhaps someone who knows how to analyze network traffic could find out, but that's beyond my skill set.

Realistically, however, what's important is that if you're having problems, you can turn off iCloud Private Relay first, and if that doesn't resolve the issue, turn off Limit IP Address Tracking. If even that's not enough, turn off Hide IP Address for Safari or Mail.

Otherwise, just leave them all on and enjoy whatever level of additional privacy they provide.

By Glenn Fleishman

Apple Adds Lockdown Mode to Protect Activists and Government Targets

Apple has announced that a new Lockdown Mode will be coming to iOS, iPadOS, and macOS later this year to deflect attempts by government-level spyware to infiltrate the devices of targeted activists, protestors, journalists, and politicians. It's the latest measure taken by Apple in its significant efforts to protect its users' devices from being compromised and data taken without their knowledge or consent.

Lockdown Mode targets key points of weakness on always-connected devices that can connect to arbitrary servers around the world and be reached through Apple services, like Messages and FaceTime.

With Lockdown Mode enabled, you won't be able to receive anything but images and text via iMessage and MMS; all other document types will be blocked. Messages also won't preview links, among other not-yet-disclosed clampdowns. Plus, FaceTime and other Apple services that can receive arbitrary inbound invitations and requests will block everyone with whom you haven't previously initiated a call or request. (The block doesn't appear to cover anyone you've talked with before, only your outbound connections. Otherwise, someone who idly contacted you in the past as cover for a future hack would be permitted, among other scenarios.)

Those changes would have protected against the most recent severe exploit for Apple devices—a *zero-day* flaw, one that was exploited in the wild before Apple knew about it—which came through maliciously formed PDFs that bypassed Apple's 2020 addition of sandbox security to Messages (see "BlastDoor Hardens iMessage Against Malware Assaults," 4 February 2021).

Apple said that Lockdown Mode also prevents browsers from executing certain "complex web technologies," a reaction to past exploits and concerns about likely future vectors. Users can opt to add sites to a trusted list. In Lockdown Mode, wired connections are blocked whenever an iPhone is locked; Apple didn't mention iPads.

Lockdown Mode also blocks the installation of configuration profiles, which some attackers have exploited to intercept all data entering and leaving a device. Nor can devices in Lockdown Mode be enrolled into mobile device management, eliminating another known worry.

Apple has focused Lockdown Mode on users facing a known risk because it will limit some of what you can do on your devices. Even by that measure, tens of millions of Apple users worldwide could potentially benefit from enabling Lockdown Mode.

Consider, for instance, Russia's aggressive arrests of any resident insufficiently patriotic regarding its unprovoked invasion of Ukraine. Suddenly, many people in a nation of nearly 150 million might have reason to worry about state-sponsored spyware. Likewise, in America, the Supreme Court's recent decision overturning Roe v. Wade, coupled with law enforcement's behavior in past abortion-related cases, has privacy advocates concerned that police may seize the text messages and digital activity of pregnant people seeking medical care.

Alongside Lockdown Mode, Apple also announced a $10 million grant to the Dignity and Justice Fund, a grant-making organization formed within the Ford Foundation's New Venture Fund. Awards from this pool of money will fund "approaches to help expose mercenary spyware and protect potential targets." This could include outreach and awareness, bolstering the security of human-rights organizations, and directing research into spyware. Apple said the fund will be advised by a committee of human-rights-oriented technologists drawn from groups like Amnesty International and Citizen Lab. Ivan Krstić, Apple's head of security engineering and architecture, will also serve on the committee.

Lockdown Mode marks both an evolution and an escalation in Apple's battle to balance compliance with legal government actions with illegal, extrajudicial, and criminal efforts to penetrate iPhone, iPad, and Mac security. Last year, Apple sued a major global government spyware supplier, NSO Group, for its alleged role in exploiting Apple products and services (see "Apple Lawsuit Goes After Spyware Firm NSO Group," 24 November 2021).

Not all potential uses of spyware are inappropriate. NSO Group and similar companies consistently claim that the primary use of their products is to combat terrorism and human trafficking. Yet there's no evidence of what Apple calls "mercenary spyware" aiding in such, while NSO Group's Pegasus has allegedly been used to target journalists in Mexico, politicians in the Catalonia

region of Spain, and human-rights activists in the United Arab Emirates, among others.

When Apple announced its lawsuit against NSO Group last year, the company said it would give $10 million to Citizen Lab and Amnesty Tech, part of Amnesty International, along with any damages it received if the company prevailed in its lawsuit. This new announcement doesn't reference that grant, even though it notes the same sum and says damages from the lawsuit will be part of the pot it donates to the New Venture Fund. Apple should provide more clarity here—are these separate initiatives, or did the previous grant get folded into this new approach?

Lockdown Mode arrives in iOS 16, iPadOS 16, and macOS 13 Ventura later this year. Enabling it appears to require just a tap, a confirmation, and a restart. You can turn it off with the same simplicity.

Should you enable Lockdown Mode even if you're unlikely to be targeted by state-sponsored actors? We suspect it will be overkill for nearly everyone and quickly become frustrating if you're not worried about your personal safety. But as far as we can tell, there's no harm in enabling it as a test or for temporary protection.

We'll see how effective Lockdown Mode ends up being in the real world, but based on the early description, Apple has set a new high-water mark in its commitment to security in the unequal battle between governments and individuals.

Apple Updates

Security Update 2022-005 (Catalina 10.15)

Jul 20, 2022 — 1.63 GB

macOS Catalina Security Update 2021-005 (19H2026) is recommended for all users and improves the security of macOS.