Zero-Trust Security for Webex

webex by CISCO

Contents

Setting the standard for confidential end-to-end encrypted meetings

Millions of people globally use Webex for meetings. As one of the most trusted online meeting platforms for large enterprise and government customers, everything from school classroom meetings to doctor/patient visits, to legislative sessions and votes take place every day on Webex.

As the demand to work remotely explodes, and global geographically disparate teams and restrictions on in-person gatherings increase, so does the need to conduct confidential meetings using collaboration tools. Webex has been meeting that need longer than any other collaboration vendor.

Webex has a long history of supporting strong End-To-End (E2E) encryption as an optional feature for meetings and always-on end-to-end encryption for messaging. Zero-Trust Security builds on this heritage in three key ways:

- Establishing a foundation of standards-based, formally-verified cryptography
- Augmenting end-to-end encryption with end-to-end verified identity
- Extending support to current Webex® devices as well as the Webex App

This white paper provides some background on end-to-end security and lays out the technical underpinnings of how our Zero-Trust Security works.

A 12-year heritage of delivering on E2E security

Webex has been doing E2E encryption for a long time. Webex Meetings was first upgraded to support end-to-end encryption in 2008—more than 12 years ago.

"End-to-end" security protects users' content from the time it is sent by a user to the time it is received and displayed by other participants in the meeting, so that entities in the middle can't read it or tamper with it. Basically, E2E security prevents your cloud conferencing provider (and everyone else) from listening to your meetings. Obviously, this takes some work; Webex has to provide you a conferencing service without being able to see the conference.

Figure 1. Levels of E2E security

Depending on how much investment a video conferencing provider makes in E2E security mechanisms, they can provide varying levels of protection, as shown in Figure 1. Any E2E security is better than just encrypting in transit and at rest.

Millions of people globally use Webex for meetings.

Most applications today that claim to provide E2E security only provide protection from passive attacks. Passive-only protections are still valuable since they make it so that bad actors can only eavesdrop on a meeting while it is ongoing. The best approach, however, is to protect against both passive and active attacks, where the conferencing provider itself might pretend to be a legitimate participant in order to enter the meeting. For this, you need an E2E identity layer in addition to E2E encryption.

With Zero-Trust Security, Webex brings new, stronger cryptography to meetings, and augments end-to-end encryption with end-to-end verified identity, to keep your meetings safe from the full spectrum of attacks.

The end-to-end user experience

Before we dive into technical details, let's walk through what it's like for a user to participate in an end-to-end secure meeting. We'll show how things look on a Webex device, but the experience in the Webex App is similar. To create an E2E encrypted meeting, the person scheduling the meeting selects the appropriate meeting type when scheduling (Figure 2).

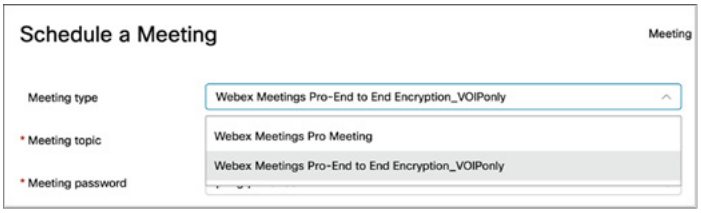

Figure 2. Scheduling an E2E encrypted meeting

When participants join the meeting, they'll see a "shield with lock" icon that indicates they're in an E2E encrypted meeting. They can tap on that icon to get more information about how the meeting is protected (Figure 3).

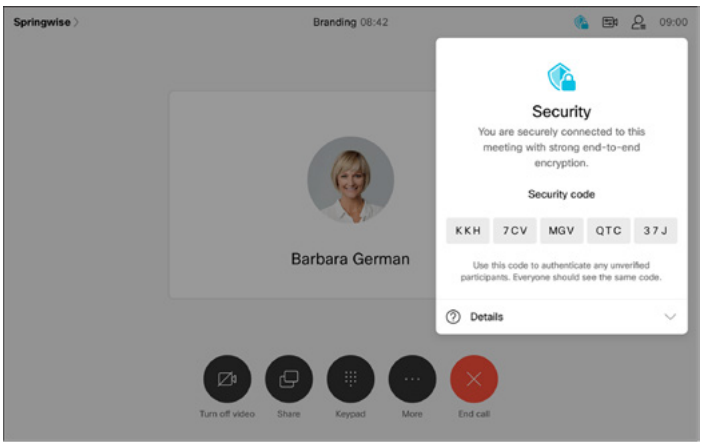

Figure 3. Security information within an E2E encrypted meeting

Finally, in the meeting participant list, the Webex application indicates whether the participant's identity has been verified in an E2E-secure way; that is, whether someone other than Cisco has vouched for the participant's identity (Figure 4). The participants can also check that they all have the same security code to confirm they all have the same view of the meeting, and are thus secure from impersonation attacks.

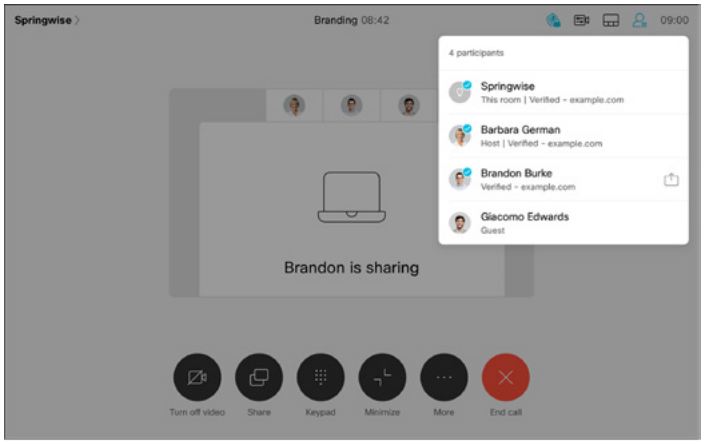

Figure 4. End-to-end identity verification

Standards-based cryptography

Zero-Trust Security is built on industry-standard cryptographic protocols, which Cisco is driving together with other leaders in security and privacy, including Google, Cloudflare, Wickr, and Wire. Cisco led the first generation of secure Voice over IP (VoIP) with technologies like the Secure Real-Time Protocol (SRTP), DTLS-SRTP, STIR/SHAKEN, and WebRTC, and we're doing the same thing today with Zero-Trust Security (Figure 5).

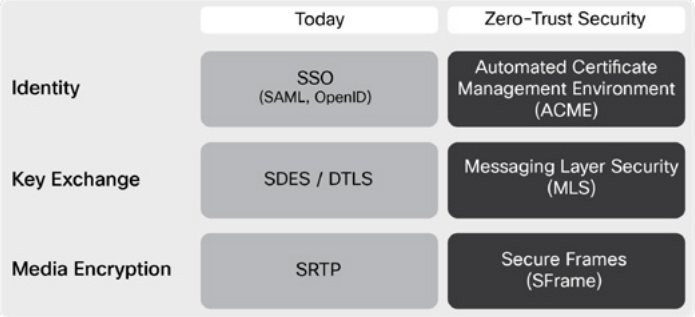

Figure 5. Standards for Zero-Trust Security

There are basically three layers to the system, and at each layer we are taking the old hop-by-hop technology and bolstering it with an additional end-to-end layer of protection. The three layers include:

Identity: E2E identity requires that clients have credentials that prove their identity. The Automated Certificate Management Environment (ACME) protocol is used to issue more than half of all web certificates today. Together with some extensions to leverage enterprise identity systems, ACME allows Webex to provide high-grade identity assurance with a seamless user experience.

Key exchange: Clients participating in an E2E meeting need to set up keys for E2E encryption without the conferencing provider being able to access those keys. The Messaging Layer Security (MLS) protocol is an emerging standard that builds on technologies developed in the open-source community and puts them in a rigorous, formally-verified setting.

Content protection: Finally, the actual media content of the meeting needs end-to-end protection. SFrame (for "Secure Frames") is a fast and simple encryption framework for encrypting real-time media, which allows an extra layer of encryption to be added with minimal overhead.

Let's look at these technologies in a bit more detail, starting from the bottom up.

Lightweight media encryption

A media encryption scheme is a way to take a key shared by the participants in a meeting and use it to protect the actual content shared in the meeting, like audio, video, and screen share. At the protocol level, the encryption scheme needs to send some extra information with each unit of encrypted data, most importantly: (1) a header that tells the receiver how to decrypt the packet; and (2) an authentication tag that the receiver uses to verify that the packet hasn't been tampered with. For example, in SRTP the unit of encryption is an RTP packet; SRTP relies on the RTP header for its setup and appends an authentication tag to the end of the packet.

Since all of this extra information increases the bandwidth required for an E2E encrypted meeting, SFrame provides the minimal amount of framing necessary. The SFrame header indicates which key should be used (allowing for key rotation) and provides a unique sequence number that defines the initialization vector. This is followed by the content encrypted with the E2E key and an authentication tag. The SFrame header, content, and tag are then wrapped in an SRTP packet, so that SRTP protects against inspection or modification by network attackers, and SFrame protects against the conferencing provider's media elements (see an example in Figure 6).

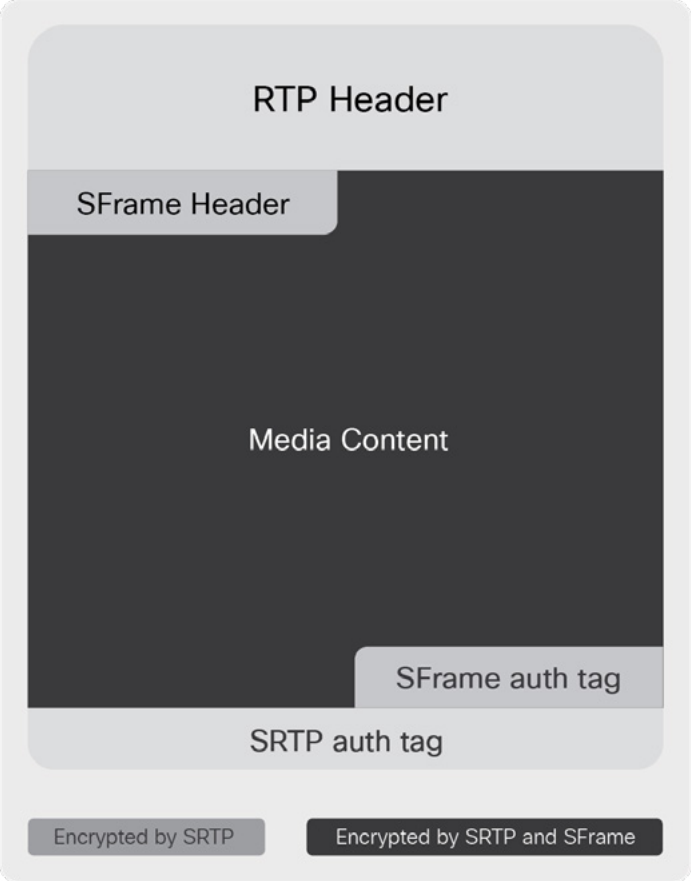

Figure 6. SFrame media encryption

This minimalist approach has a couple of other benefits: SFrame is independent of the underlying transport, so it's forward compatible with emerging technologies like RTP over QUIC and RIPT. It is also possible to configure SFrame so that it works on multi-packet units (e.g., whole frames) to further reduce the encryption overhead.

Group key exchange

Setting up a secret key among a group when the group only has access to an untrusted channel seems magical. Techniques for doing this were first developed in the 1980s, and today, more than 90 percent of web browsing uses Transport Security Layer (TLS) to do exactly this. Messaging Layer Security (MLS) extends this functionality to groups (TLS only works point to point), so we can use it to set up conference keys, even when the conference provider is untrusted.

MLS provides a few critical security features. It:

- Establishes a shared key that is known only to authenticated members of the group
- Assures that all members of the group have a consistent view of who is in the group
- Rotates the shared key when a member joins or leaves the group

The first feature provides us with the keys we use to encrypt media with SFrame, and together with the second feature, it provides the starting point for E2E identity. The third feature ensures that the only people who can decrypt the media are the people who are in the meeting right now—you can't decrypt media from before you joined or after you left (or were kicked out).

To accomplish all this, the clients in a meeting exchange MLS messages at critical points in the meeting, in order to maintain an MLS group that contains exactly the participants in the meeting at any point in time. Webex routes the messages among the clients, but MLS protects these messages so that it's safe for Webex to handle them. Figure 7 shows how the messages flow.

Creation: The first person to join the meeting (host

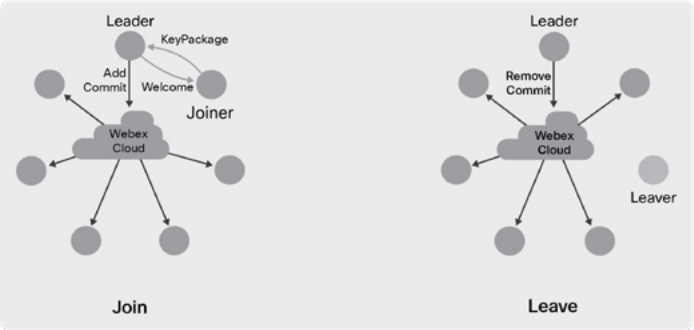

Figure 7. Group key management with MLS

or not) creates the MLS group for the meeting and becomes the "leader" for purposes of MLS. At any given point in time, a single participant in the meeting is designated as the leader; in most cases, this will be the meeting host. Once the session is started, MLS assures that every meeting participant has the cryptographic information to act as leader; Webex simply designates one to avoid confusion.

Join: When a new participant wants to join the meeting, they ask to be let in by sending an MLS KeyPackage message to the leader, which provides their public key and authentication information. The leader responds with a Welcome message that tells the joiner about the other participants in the meeting. The leader also broadcasts an Add message and a Commit message to the group. The Add message announces the new joiner to those already in the meeting. The Commit message causes all the current participants to hash the old group key to get a new key for the now-extended group; the Welcome message provides this key to the joiner.

Leave: When someone leaves the meeting, it's the leader's job to lock them out. The leader does this by broadcasting MLS Remove and Commit messages to everyone remaining in the meeting. The Remove message tells the remaining participants who left and the Commit encrypts a new key to all the remaining participants (but not the one who left). The remaining participants hash the new key with the old group key to get a key that the leaver doesn't have. If the participant who left is the leader, then Webex will appoint a new leader and the new leader will remove the old leader.

Note that both joins and leaves result in MLS generating a new key for the group. Whenever this happens, the participants in the meeting will update to this new key within a few seconds. MLS can also gracefully handle multiple Add and Remove transactions in one Commit, to avoid a flurry of unnecessary, expensive key rotations in situations such as the start or end of a meeting. After each rotation, the participants in the meeting delete their copies of the old keys so that even if one of them is compromised, the old meeting content is safe. Finally, at the end of the meeting, the participants delete any remaining keys, so that even if the content of the E2E meeting was stored somewhere, it can no longer be decrypted.

Key rotation can mean that it takes a couple of extra seconds to join a meeting or kick someone out, but it gives clear security assurance: If someone doesn't appear in the participant list for the meeting, they don't have the keys to decrypt the content.

End-to-end verified identity

End-to-end identity verification requires credentials that are anchored outside of Cisco so that Cisco can't tamper with them. We can provide that level of assurance for customers who are willing to put some additional mechanisms in place, but we also provide some lower-assurance mechanisms that apply for all users and all meetings.

MLS uses standard X.509 certificates for authenticating the participants in a meeting. A certificate is a signed statement by a trusted "certificate authority" that the holder of a given signing key has a given identity. Each participant in a meeting gets a certificate in one of the following ways:

For clients in SSO-enabled organizations: When a user logs into Webex using SSO, their client also uses their SSO login with an extension to ACME to prove the user's identity to a trusted certificate authority, which issues them a certificate based on that proof. The whole process can occur behind the scenes, without user interaction, since ACME automates the whole process. A client that goes through this flow uses its certificate to prove its email address to other clients.

For devices configured for E2E by an organization's admin: Cisco already provides administrators a tool to provide a domain name to a Webex device. To support E2E identity, Webex has extended this tool to automatically acquire a certificate for the device using the standard ACME process for domain names. The device uses this certificate to prove to other participants in a meeting that it legitimately represents its domain name.

Otherwise (guests, non-SSO users, non-provisioned devices): By default, a Webex client or Webex device gets a certificate from a Webex certificate authority, which reflects the identity that Cisco has verified in a way that can be presented to other clients.

The participant list in an E2E encrypted meeting shows the authentication status of each user. The first two of the options above provide true E2E verified identity, in the sense that Cisco cannot tamper with it. In these cases, the participant list shows the domain that the participant belongs to. For example, in the meeting shown in Figure 8, you can see that Barbara and Brandon have verified email addresses from example.com, and the Springwise room system has a verified domain name under example.com. Certificates from Webex provide a more limited protection—only an attacker that compromises the Webex certificate authority can impersonate a user. Even though this is better than having no identity protection at all, such participants in the meeting are not marked as verified.

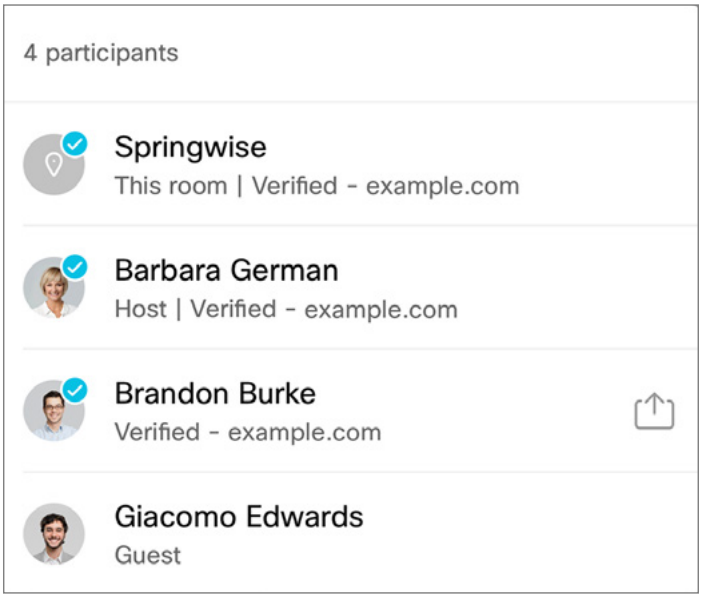

Figure 8. Verified identity visible in the participant list

As a backstop to these procedures, Webex also provides a “security code” for the meeting. A security code is a short, human-readable string computed as an output of the MLS process. The security code summarizes the cryptographic state of the meeting, including the group’s keys as well as the certificates presented by the participants. This code can be used to detect an impersonation attack because such an attack would result in at least one participant having a different cryptographic state than everyone else. But it only rules out impersonation if everyone verifies that they have the same code. If some subset of the participants verifies that they have the same code, they know that their identities are faithfully represented to each other, but someone who doesn’t confirm their code could have been impersonated. Figure 9 shows an example of how the security code is displayed in a meeting.

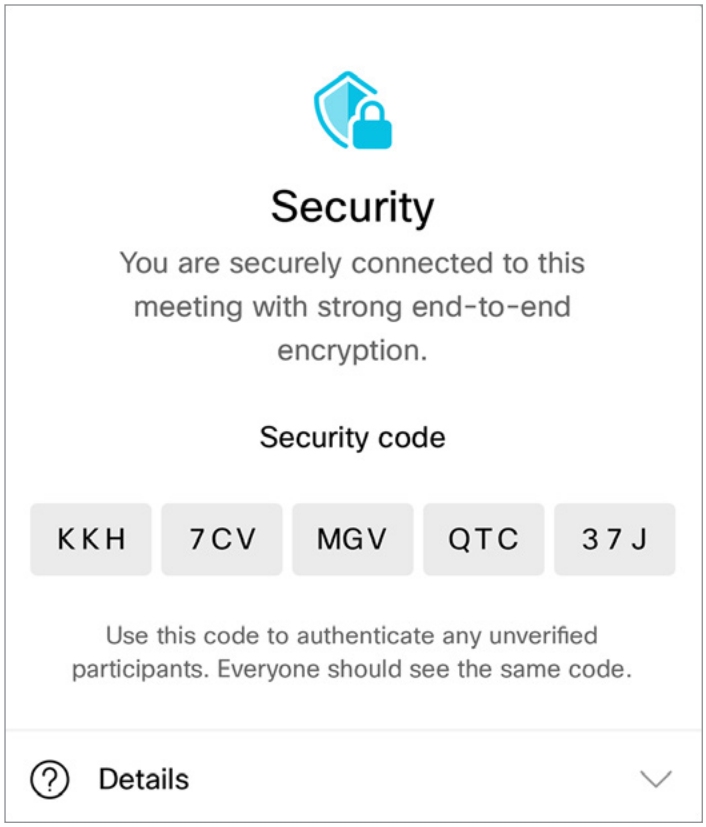

Figure 9. A security code to protect against impersonation

This code provides a very strong defense against impersonation attacks. In fact, it is part of the MLS protocol so it has had robust security analysis. Because it is tightly bound to the state of the meeting, the security code changes every time someone joins the meeting. As a result, participants in the meeting will need to re-verify their security codes every time someone joins. (The code doesn’t rotate when someone leaves, because leaving can’t introduce an impersonator.)

Summary

Zero-Trust Security was launched in the first quarter of 2021, with some big improvements to the E2E security of Webex meetings:

- Standards-based, formally-verified cryptography
- E2E verified identity
- Support for all Webex devices available for sale
- Support for E2E encryption in Personal Meeting Rooms

Like the current feature, it will have some limitations. E2E encrypted meetings do not support:

- The Webex App
- Older Webex devices, including the SX, DX, and MX series
- Features provided by Cisco cloud services that require access to plaintext media, including:
  - Network-Based Recording (NBR)
  - Saving session data, transcripts, meeting notes, etc.
  - Remote computer sharing
  - PSTN or SIP interoperability

Future directions

Going forward, we will improve E2E encrypted meetings in a few dimensions, guided by our principles of granting the cloud the least privilege necessary and giving customers control over their data.

Integrating customer-authorized services: Most of the features that are disabled in E2E encrypted meetings right now are disabled because they are provided by processing media on a Cisco operated server. Obviously, if an E2E encrypted meeting is going to lock out Cisco, we can't let these servers into the meeting! But we realize that some customers might want a different trade-off between security and functionality, so we're working on tools to let customers navigate that trade-off by selectively integrating services as "ends" in the meeting. One advantage of our open, standards-based approach is that it makes it easy for partners and other vendors to integrate with Webex's E2E encryption, so that customers can bring in third-party services in addition to services that Cisco might offer.

To make sure that all this flexibility doesn't undermine the security of E2E encrypted meetings, we'll provide customers with a robust policy framework to define what types of services are allowed in what contexts. For example, a company's IT administrator might define "high-security" and "medium-security" meetings, where high-security meetings are totally locked down (and thus missing features) and medium-security meetings allow a handful of authorized services to be added.

Decentralized identity: Our current SSO-based strategy enables us to provide E2E verified identity very broadly, for any SSO-enabled user, but ultimately, we'd like to allow customers to bring their own E2E certificate authorities. This provides better security for everyone, since there are fewer entities involved in authenticating meeting participants, and customers can really take control of their meetings' security. In other words, our goal is a decentralized identity ecosystem.
And the backbone of any decentralized system is standards. While some additional standards will be necessary to complete the decentralization story, the technologies we're putting in place now provide the foundation for interoperability between E2E identity authorities and the clients that need certificates.

Ubiquitous E2E security: Everything should be E2E secure—not just select meetings. The first step to achieving this vision will be to enable E2E security for any endpoint that can join a Webex meeting. The obvious first candidates are the Webex App and cloud-connected Cisco phones, and eventually, our open, standards-based E2E architecture should enable us to bring in Session Initiation Protocol (SIP) infrastructure as well.

The innovations discussed in this paper show a clear path to our long-term vision of always-on E2E encryption. Once users can connect from anywhere and customers can securely opt in to the features they want in an meeting, we'll be able to use E2E encryption for every Webex meeting—from informal ad-hoc meetings with all the features turned on, all the way up to locked-down, high-security meetings. E2E encryption will always assure that the meeting's content is only available to authorized participants, and customers and users will decide what set of participants is right for a given meeting.

Conclusion

As customers continue to hold more confidential meetings over video conferencing tools, Webex is defining the future of end-to-end secure meetings with standards-based cryptography, end-to-end verified identity, and availability across both Webex applications and devices. Our open, standards-based approach will allow us to extend end-to-end protections to cover a range of scenarios, in cooperation with an ecosystem of partners.

For more information
Please visit webex.com

June 2021

webex by cisco