GUIDE

Communication Security Guide for Web3 Teams

Choosing channels by sensitivity, hardening the ones you use, and confirming out-of-band

Prepared for	Web3 teams	Classification	Public
Prepared by	Oak Security	Date	2026-06-17

Oak Security GmbH

1. What This Guide Is For

This guide is for everyone on a Web3 team who sends a message that could move money, grant access, or leak a plan: signers, operators, engineers, finance, founders. Most of it does not require deep technical knowledge. It requires you to pick the right channel for what you are saying, to harden the channels you use, and to confirm anything consequential on a second channel before you act.

The aim is not to make the team unreachable. It is to make sure that one intercepted message, one impersonated contact, or one ported phone number does not become a treasury loss or an admin takeover.

> **NOTE**
>
> **Baseline rule:** match the channel to the sensitivity of the message, verify who you are talking to, and confirm any financial or access-related action out-of-band before you act on it.

No single tool is correct for everything. Every channel is correct for some uses and wrong for others. The rest of this guide is about telling them apart.

2. Choosing a Channel by Sensitivity

A useful frame is **CIANA**: Confidentiality (only the intended recipient reads it), Integrity (it arrives as sent), Availability (it works when needed), Non-repudiation (the sender cannot later deny it), Authenticity (the sender is who they claim). Plain email fails confidentiality and authenticity by default. Telegram fails authenticity. Signal passes most of CIANA once safety numbers are verified.

Channel	Sensitivity level	What it is for	What it is not for
In person, device-free	Highest	Seed phrases, recovery material, signing-ceremony coordination, anything catastrophic if recorded.	Anything that needs a record.
Signal (verified safety numbers)	High	Sensitive operations, incident communications, treasury coordination, high-trust one-to-one and small groups.	Broad team broadcast, long-lived records.
Encrypted email (S/MIME or PGP)	Medium to high	Sensitive contracts and formal correspondence that needs confidentiality plus a record.	Casual chat. Clunky tooling; key management is the weak point.
Email with SPF, DKIM, DMARC	Medium	Contracts, formal correspondence, legal-record uses. Authenticity and non-repudiation, not confidentiality.	Authorizing any high-value action. Confidential content in the body without encryption.
Slack / Microsoft Teams (enterprise)	Low to medium	Team collaboration, internal records, day-to-day coordination.	Secrets, signing material, anything catastrophic if a misconfigured external integration logged it.
Discord	Low	Community	Anything signing-

		engagement, public channels.	related. Weaker enterprise controls than Slack.
Telegram	Lowest	Casual chat only.	Keys, secrets, transaction signing, admin decisions. Non-E2E by default, impersonation trivial, no enterprise controls.
SMS	Avoid for security	Nothing security-relevant.	Two-factor codes, recovery, any account of value. See Section 7.

Why Telegram Is Wrong for Sensitive Ops

Telegram chats are not end-to-end encrypted by default. Only "Secret Chats" are, and they are off by default, one-to-one only, and not available in group chats. Account impersonation is trivial: a new account with a copied display name and photo is enough to fool a busy operator. There are no enterprise audit controls. Use Telegram for casual chat and community presence. Never use it for keys, secrets, signing, or admin decisions.

Where Signal Fits

Signal gives end-to-end encryption with forward secrecy, verifiable safety numbers, and minimal server-held metadata. It is the right default for sensitive operations, incident response, and high-trust small groups. The condition is that you verify safety numbers before discussing anything that matters (Section 4). An unverified Signal contact is an assumption, not a guarantee.

The Limits of Encrypted Email

Encrypted email (S/MIME or PGP) gives confidentiality plus a durable record, which Signal does not. PGP remains the standard for encrypted email and signed git commits. The tooling is awkward and key management is where most people get it wrong: an expired key, a wrong key, or a private key on a compromised laptop defeats the encryption. Encrypted email is for sensitive formal correspondence that needs a record, not for fast-moving operational chat.

3. Securing the Channels You Use: Signal

Choosing Signal is half the work. A misconfigured Signal account leaks the things you moved to Signal to protect.

Required Configuration

Setting	What to do	Why
Registration Lock	Turn on Settings > Account > Registration Lock.	Requires your Signal PIN to re-register the number on a new device. Blocks an attacker who has SIM-swapped you (Section 7) from hijacking your Signal account.
Signal PIN	Set a strong PIN, not a birthday or a repeated digit.	The PIN backs Registration Lock and recovers your profile. A weak PIN undoes the lock.
Disappearing messages	Set a default disappearing-message timer for sensitive conversations.	Limits how much sensitive history sits on devices that can be lost, seized, or compromised later.
Screen lock	Turn on app-level screen lock (Settings > Privacy > Screen Lock).	A locked phone with an unlocked Signal app still leaks everything.
Screen security	Turn on screen security to block screenshots and hide content in the app switcher.	Reduces accidental and malicious capture of message content.
Linked devices	Review Settings > Linked Devices regularly. Remove anything you do not recognize.	A linked desktop you forgot about is a full copy of your messages.

NOTE

Hard rule: Registration Lock on, with a strong PIN, for every team member who uses Signal for sensitive work. This is the single control that survives a SIM-swap.

4. Verifying Contacts

End-to-end encryption protects the message in transit. It does nothing if you are talking to the wrong person. Verification is how you know the person on the other end is who they claim.

- **Signal safety numbers.** Open the conversation, tap the contact name, view the safety number. Compare it with the other person over a separate trusted channel (in person, or a video call where you already know the face and voice) or scan their QR code in person. Once verified, Signal marks the contact. If the safety number later changes, Signal warns you. A changed safety number means a new device or a reinstall, and it means re-verify before discussing anything sensitive.
- **Treat a changed safety number as suspicious until explained.** It is often benign (new phone), but it is exactly what an account takeover also looks like. Confirm the reason out-of-band.
- **Verify before, not after.** Verify the contact before the conversation that matters, not in the middle of it.

> **NOTE**
> **Hard rule:** verify the Signal safety number of every counterparty before discussing keys, treasury actions, signing ceremonies, or incident details.

5. Out-of-Band Confirmation for Financial and Access Actions

The single most common shape of a large loss is a message that looked like it came from a trusted person, asking for a transfer or an access grant, acted on without a second check. Email, chat, and even a video call are not proof of identity. A live deepfake on a camera-on call is now operational; one engineering firm authorized roughly $25 million across fifteen transfers in February 2024 to a call where every "colleague" was synthetic.

The defense is procedural, not visual. For anything that moves funds or grants access, confirm on a second channel the requester did not choose.

Action	Out-of-band confirmation
Treasury transfer or wire	Callback to a known number stored in advance, not the number on the screen. Dual signoff, second signer reached on a different channel from the first.
Multisig signature request	Coordinate over Signal with verified safety numbers. Verify calldata independently; do not trust the request message.
DNS, registrar, or domain change	Voice or in-person confirmation with the named owner. Never authorize from an unsigned email.
Access or permission grant (deploy rights, IAM, multisig seat)	Confirm with the requester through a pre-agreed channel before granting.
Any "urgent" request from an exec or counterparty	Pre-agreed challenge phrase, agreed in a different channel before the call. Out-of-band callback.

NOTE

Hard rule: email must never be the channel that authorizes a high-value action. A treasury wire, a multisig signature, or a DNS change does not happen on an unsigned email.

The strongest defense against a manufactured-urgency ask is a culture where ″let me check and come back″ is encouraged, not punished. Attackers manufacture urgency because urgency wins. A team that visibly slows down stops most of these at the ask.

6. The OpSec Circle of Trust and Team Hygiene

The **OpSec circle of trust** is deliberately small: for any sensitive action, the set of people who need to know is named, small, and bounded. Information leaves the circle only when there is a reason. Every additional person in the loop is another channel that can leak, another device that can be compromised, another identity that can be impersonated.

Daily Habits

- Discuss sensitive decisions in named channels with verified members, not in broadcast spaces.
- Coordinate treasury actions among signers, not in company-wide Slack.
- Agree signing-ceremony details over Signal with verified safety numbers.
- Keep the team's threat model internal. An attacker who knows your threat model has a head start.

Who Can Be Impersonated

Anyone whose name carries authority to request money or access is an impersonation target: the CFO, a founder, a senior engineer, a portfolio-company contact. The more public a person's role and voice, the cheaper they are to clone. Map who on your team can authorize consequential actions, and put the out-of-band confirmation from Section 5 around every one of those roles.

Insiders Are Real

The circle of trust also bounds insider risk. Grant access to the task, not the role: no engineer gets deploy rights automatically, no operations hire gets a multisig seat automatically. Enforce a two-person rule on production actions (merge and deploy are different people; multisig on treasury and admin). Treat offboarding as a security event and rotate every secret the departing person touched: tokens, cloud credentials, multisig seats, SSH keys, chat tokens. A former employee who still holds publish or signing rights months after leaving is a live attack surface.

7. SIM-Swap Defense

A **SIM-swap** is the textbook crossover attack. The attacker social-engineers your mobile carrier (by phone, in a retail store, sometimes through a bribed employee) into porting your number to a SIM they control. Once they own the number, every SMS code and SMS-based reset funnels to them: email recovery, exchange logins, banking 2FA, password-manager recovery. Anything gated on the phone number is now gated on the attacker. The Michael Terpin case (2018, roughly $24 million) is the canonical example, and the pattern has only become routine since.

Defense	What to do
Carrier PIN and port-out lock	Turn it on for every line in the team and family. Every major carrier supports it. Re-check annually; carriers reset it during account migrations.
Move 2FA off SMS	Use TOTP authenticator apps or hardware keys (YubiKey, Titan). Nothing of value gated on SMS: not exchange 2FA, not email recovery, not banking, not password-manager recovery.
eSIM where possible	Physical SIM slots are the most common swap channel. eSIMs require additional carrier verification to swap.
Carrier choice	Postpaid beats prepaid. Business lines with a named account manager beat consumer lines. MVNOs are historically weaker on port-out defense.
Treat a dead number as an emergency	If SMS, calls, or data suddenly stop, assume a port-out is in progress. Call the carrier from a different line immediately. Minutes matter.

> **NOTE**
>
> **Hard rule:** no account of value uses SMS for two-factor or recovery. Move every one of them to TOTP or a hardware key, and lock the carrier account.

The SIM-swap link to Section 3 is direct: Signal Registration Lock is what stops a successful SIM-swap from also taking over your Signal account.

8. Metadata, Links, and Attachments

Content is not the only thing that leaks. **Metadata** (who talked to whom, when, how often, file authorship and EXIF data, location tags) reveals structure even when the message body is encrypted. Signal minimizes server-held metadata; most other channels do not. Strip metadata from files before sharing externally: document authors and revision history, image EXIF and GPS, PDF producer fields.

Link and Attachment Hygiene

The payload in a targeted attack arrives as something runnable or clickable: a repo to clone, a PDF to open, a link to "verify your wallet."

- Treat any link or attachment from unsolicited contact as hostile until verified out-of-band on a channel the sender did not introduce.
- See the full URL before following any link. Visible link text lies. Watch for look-alike domains (`oaksecurityy.io`, `oak-security.com`), and Unicode homoglyphs (Latin letters swapped for visually identical Cyrillic).
- Type URLs by hand or use bookmarks for services the team uses. Email and chat notifications exist to be ignored, not clicked.
- Never run unknown code on a work machine. Use a disposable VM or sandbox for any take-home, prototype, or "quick test." A `postinstall` script runs the moment you type `npm install`.
- Encrypt sensitive files with AES-256 (`age` or 7-Zip), not password-ZIP, which is trivially cracked. Share the decryption password on a different channel from the file. A password manager's secure-share feature is built for this.
- When distributing artifacts externally, post hashes on a separate host from the binary so a CDN compromise on the binary does not also compromise the hash.

9. Communications-Security Checklist

Per Person

Check	Done
I know which channel to use for each sensitivity level (Section 2).	
Signal Registration Lock is on, with a strong PIN.	
Disappearing messages and app screen lock are set in Signal.	
I have verified safety numbers with the people I do sensitive work with.	
Carrier PIN and port-out lock are set on my line.	
All my accounts of value use TOTP or a hardware key, never SMS.	
I confirm financial and access requests out-of-band before acting.	
I treat unsolicited links and attachments as hostile until verified.	
I run unknown code only in a sandbox or disposable VM.	

Per Team

Check	Done
Approved channels per sensitivity level are documented and known.	
Telegram is not used for keys, secrets, signing, or admin decisions.	
Sensitive decisions live in named channels with verified members.	
The OpSec circle of trust is named and bounded for each sensitive action.	
Roles that can authorize money or access are mapped, with out-of-band confirmation around each.	
Email outbound has SPF (`–all`), DKIM (2048-bit, rotated annually), DMARC at `p=reject`.	
Two-person rule enforced on production and treasury actions.	
Offboarding rotates every secret the departing person touched.	
"Let me check and come back" is encouraged, not punished.	

10. References

- Signal Registration Lock: https://support.signal.org/hc/en-us/articles/360007059792
- Signal safety numbers: https://support.signal.org/hc/en-us/articles/360007060632
- SEAL (Security Alliance) frameworks: https://frameworks.securityalliance.org
- CISA guidance on SIM-swap and SMS 2FA: https://www.cisa.gov/secure-our-world
- DMARC, SPF, DKIM overview: https://dmarc.org/overview/