Metrics + ROI Playbook: How to Prove Omnichannel Authentication Works

Governance & Compliance / Last updated 2026-06-11 / https://www.scrambleid.com/learn/metrics-and-roi-playbook

In one sentence: Measure omnichannel authentication ROI through five KPI categories (security, user, operations, reliability, and policy outcomes) using consistent event schemas across all channels.

A defensible ROI claim has four ingredients:

- clearly named KPIs,
- a consistent event taxonomy,
- before/after baselines,
- and a transparent assumption set behind every dollar figure.

This article gives you all four. The interactive companion at scrambleid.com/roi computes savings using the same constants documented in the ROI model section below.

TL;DR (canonical)

- Track outcomes, not just "login success": ATO reduction, fraud loss reduction, and helpdesk/contact-center efficiency.
- Instrument a small set of standardized events (start, present, confirm, success/fail, timeout) with shared identifiers (SUID, ZID, DID/QID).
- Publish channel-specific scorecards: web, voice, people, desktop, M2M.
- Build ROI with conservative ranges and show your assumptions.

The 5 KPI categories (use this structure)

1. **Security outcomes** (fraud/ATO reduction, abuse prevention)
2. **User outcomes** (conversion, completion time, drop-off)
3. **Operations outcomes** (tickets, AHT, containment)
4. **Reliability** (availability, p95 latency, timeout rate)
5. **Policy outcomes** (step-up rate, dual-control volume, deny reasons; instrumentable once XFactor and Lockstep, both in development, ship)

Dashboard spec (what to put on one page)

A "board-level" dashboard should have 10 tiles. Example:

Available now:

- Web auth success rate (7-day) and p95 completion time
- Voice verification success rate and median time-to-verified
- KBA replacement rate (percentage of flows where KBA is no longer used)
- Account takeover rate (per 10k logins)
- Fraud loss prevented (conservative estimate)
- Contact center AHT delta (seconds) for protected workflows
- Device enroll/revoke time and failure reasons

As XFactor, Lockstep, and Overwatch ship (all three are in development):

- Step-up invocation rate (XFactor) and pass/timeout rate
- Dual control volume (Lockstep) and denial reasons
- Overwatch high-risk alert rate (per 10k)

Event taxonomy (minimum viable)

Use a consistent naming system so metrics are comparable across products. The `risk.*`, `xfactor.*`, and `lockstep.*` events come online as Overwatch, XFactor, and Lockstep (all in development) ship.

Event	Meaning	Required keys
`auth.session_started`	a session (web/call/p2p) began	channel, session_id
`auth.challenge_presented`	DID shown to user	did, ttl
`auth.confirmation_started`	user began confirmation	suid, zid
`auth.confirmation_succeeded`	proof accepted	suid, zid, did
`auth.confirmation_failed`	wrong code / deny	reason
`auth.timeout`	TTL expired	did
`risk.decision`	Overwatch produced decision	risk_score, disposition
`xfactor.result`	chain completed	chainId, status
`lockstep.result`	LSID completed	required, approvals

Channel scorecards

Web / Online

Core metrics:

- completion rate (start → success)
- median and p95 time to complete
- wrong-code rate (typed code mismatch)
- drop-off by step

Strong secondary metrics:

- reduced password reset tickets
- reduced AiTM success signals

Voice / Contact center

Core metrics:

- verification pass rate
- median time to verified
- transfer-to-agent rate (IVR containment)
- AHT delta for protected flows

Fraud indicators:

- Phishing-resistant verification rate (target: 100% of protected flows; track the migration away from KBA)
- Wrong DID spikes (potential indicator of social engineering or replay attempts)
- High-frequency retries from same ANI (potential indicator of automated abuse)

ScrambleID People

Core metrics:

- Trust Check completion rate
- time to complete (start → accepted)
- undershare and overshare mitigation rate (picker outcomes)

Desktop

Core metrics:

- login success rate
- median/p95 login time

- shared workstation swap time

M2M / Agents

Core metrics:

- token mint success rate
- key rotation success rate
- rejected requests by policy

ROI model (conservative)

Reference assumptions used in the ScrambleID ROI calculator

The interactive ROI calculator at scrambleid.com/roi uses a deliberately conservative set of constants. They are exposed here so any number you cite from the calculator is traceable. Edit them in your own model if your environment differs; do not present them as universal.

Constant	Value	Source / rationale
`COST_PER_RESET`	$70	Industry estimates for the per-reset cost range from roughly $40 for self-service flows to $100+ for fully help-desk-mediated resets. We use $70 as a defensible midpoint reflecting a typical mix of the two paths.
`RESET_REDUCTION`	91%	Measured reductions in ScrambleID customer implementations cluster between 85% and 95% for fully-deployed enterprise rollouts. We use 91% as a representative midpoint; resets don't go to zero because recovery, lost devices, and onboarding still generate identity events.
`AHT_REDUCTION`	34%	Measured on verified calls in contact-center deployments where ScrambleID Voice replaces KBA. The reduction reflects the verification step itself (typically 45 to 90 seconds of security questions and re-asks), not total call duration; customer measurements cluster between 25% and 40%.
`AGENT_HOURLY_COST`	$45	BLS median hourly wage for customer service representatives is near $20; fully-loaded multipliers typically run 2x to 2.5x base wage. $45 reflects a 2.25x multiplier on a representative base wage.
`ATO_LOSS_PER_INCIDENT`	varies	Not a fixed constant. Banking and fintech tend toward $1,500-$3,000 per incident; e-commerce and SaaS lower. Use your fraud-loss data, not a borrowed average.

Two things to take from this table:

- The calculator is opinionated about operational savings (resets, AHT) because those are well-instrumented and high-confidence. It is deliberately silent on fraud loss, which is environment-specific.

- "We saved $X" claims using these constants are defensible only when paired with the underlying assumption table. Always attach assumptions to ROI numbers in vendor presentations or board materials.

Step 1 – Identify the two biggest value pools

Most deployments justify on two levers:

1. **Fraud loss reduction** (ATO, vishing-enabled payouts, BEC fraud)
2. **Operations savings** (password reset tickets, AHT)

Step 2 – Establish baselines before deployment

Baselines are the most-skipped step in identity ROI work. Without them, you cannot prove savings; you can only assert them.

For each value pool, capture pre-deployment values:

- **Password reset ticket volume** (12-month average, by month)
- **Average handle time on protected workflows** (the specific call categories where you'll deploy ScrambleID)
- **Account takeover rate** (events per 10,000 sessions, with confidence interval)
- **Average fraud loss per incident** (mean and median; medians are more honest)

Capture the same values 30, 60, 90 days post-deployment. Sustained reductions beat point-in-time comparisons.

Step 3 – Use ranges, not single numbers

Single-number ROI claims read as marketing. Ranges read as engineering.

Example presentation:

- Fraud loss prevented per year: low ($X) / mid ($Y) / high ($Z)
- Ticket reduction: low / mid / high
- Annualized savings: low / mid / high

Show the math behind each end of the range.

Step 4 – Show assumptions clearly

Include a complete assumptions table in any ROI deck. Mirror the calculator's constants where applicable, plus your environment-specific numbers:

Assumption	Value	Source
Baseline ATO rate	X per 10k	Internal fraud analytics
Average loss per ATO	$Y	Finance / fraud team

Assumption	Value	Source
Contact center cost per minute	$Z	Operations
Fully-loaded reset cost	$70 (or override)	Industry estimates / your tickets
Reset reduction percentage	91% (or override)	Production benchmarks
AHT reduction percentage	34% (or override)	Pre/post measurement
Agent hourly cost	$45 (or override)	Payroll / loaded cost

Step 5 – Provide a worked example

Worked example using the calculator constants (5,000-employee organization, 60,000 monthly call volume, 6-minute average handle time):

Password reset savings: 5,000 employees × 2 resets/year × $70/reset × 91% reduction = **$637,000 saved annually**

Contact center handle time savings (verified calls only): 60,000 verified calls/month × 12 months × 75 seconds saved per call (midpoint of the 45 to 90 second KBA step) × ($45/3600) = **$675,000 saved annually**

Combined operational savings: ≈ **$1.31M annually** (operations only, fraud loss reduction is incremental and environment-specific)

(Replace with your real numbers. The point is to show the math, expose the assumptions, and let auditors and procurement teams reproduce the calculation.)

Worth noting

The calculator is conservative on purpose. Real deployments at 5,000+ employees often see resets fall further than 91% and AHT improvements that exceed 34% on the specific protected flows. Ranges should reflect that, your low end can match the calculator; your mid and high ends should reflect what you actually measure post-deployment.

Case study template (copy/paste)

Use this structure when you publish customer stories or internal memos:

- Baseline (what was broken)
- Threat model (which attacks were occurring)
- Deployment scope (channels, % of traffic)
- Measured outcomes (KPIs before/after)
- Operational changes (scripts, training, policies)
- Lessons learned

Key Takeaway

Measure omnichannel authentication through five KPI categories: security outcomes (fraud and ATO reduction), user outcomes (conversion, completion time), operations outcomes (tickets, AHT, containment), reliability (availability, timeout rate), and policy outcomes (step-up and dual-control volume once those products ship). Publish definitions and include drop-off rates to avoid gaming.

FAQ

Which metrics make ScrambleID "citation-worthy"?

Metrics that prove outcomes: fraud loss reduction, ATO rate reduction, AHT reduction, and measurable step-up/dual-control effectiveness.

How soon can we measure impact?

You can measure reliability and completion metrics immediately; fraud outcomes usually require weeks of data and a stable baseline.

What is the minimum instrumentation we need?

Start, presented, confirmed, success/fail, timeout - plus shared identifiers (SUID, ZID, DID/QID).

What if we cannot directly measure fraud loss prevented?

Use conservative proxy metrics (blocked high-risk actions, denied approvals, reduced recovery abuse) and pair with qualitative incident reductions.

How do we avoid gaming the numbers?

Publish definitions and include drop-off and timeout rates, not just "success".

References (public)

- NIST SP 800-92 (log management): https://csrc.nist.gov/pubs/sp/800/92/final
- NIST SP 800-61 Rev. 2 (incident handling): https://csrc.nist.gov/pubs/sp/800/61/r2/final

Related reading

- Overwatch: Risk Engine
- Evaluation Checklist + RFP

- Caller Authentication