SYSTEMS SECURITY ENGINEERING

Trustworthy Secure Design

Engineering Security from the Inside Out

Dr. Ron Ross

CEO, RONROSSECURE, LLC

“The most consequential security decisions are often made before a security control is selected.”

THE CENTRAL ARGUMENT

Security Must Be Engineered Into the System

The operational problem

Security controls remain essential.

But controls cannot compensate for weak architecture.

AI compresses attacker timelines and raises the cost of structural weakness.

The design question

What system behaviors, states, and conditions can lead to unacceptable loss—and how can the design eliminate or bound them?

Begin with loss and system behavior—not with a catalog of controls.

INTEGRATED STRATEGY

Key Concepts Shape Trustworthy Secure Design

1 Design for Intended Behavior

Define authorized outcomes, system states, and conditions that can lead to loss.

Security is a system property—not an add-on feature.

2 Use an Order of Precedence

Eliminate or reduce loss potential through design first. Then add engineered features, visibility, procedures, and training.

Design principles, life-cycle processes, and assurance turn the strategy into disciplined engineering.

START WITH LOSS

Behaviors, States, Conditions, and Loss

Define the problem before committing to a security technology.

INTENDED BEHAVIOR

What must the system do?

INTENDED STATES

What states reflect that behavior?

LOSS STATES

What states can produce unacceptable loss?

ALTER DESIGN

Prevent loss and limit what remains.

ITERATE

Assume protective functions can fail.

Do not begin by asking which control to add. Ask *why* the system permits the unacceptable loss to occur.

BALANCED PROTECTION

Preemptive + Reactive by Design

Perfect prevention is not a credible design assumption.

PREVENT

Remove avoidable loss paths and conditions.

BOUND

Limit privilege, sharing, and propagation.

DETECT

Recognize dangerous states in time.

CONTAIN

Preserve boundaries when prevention fails.

RECOVER

Restore to an authorized or secure state.

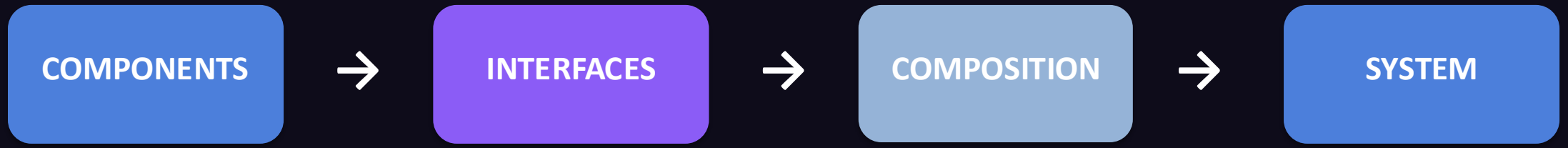
ADAPT

Revisit assumptions as conditions change.

EMERGENT SECURITY

Trustworthy Components Do Not Guarantee a Trustworthy System

System-level behavior emerges from components AND their interactions.



Reduce unwanted coupling

Domain Separation • Least Sharing • Reduced Complexity

Control necessary interaction

Mediated Access • Least Privilege • Clear Abstractions

Trustworthy secure components are necessary. Trustworthy secure composition determines whether the system behaves as intended.

SECURITY DESIGN ORDER OF PRECEDENCE

Engineer the Loss Out Before You Manage It

THE PRIORITY

Minimize the loss potential built into the system before relying on operational compensation.

Earlier alternatives generally provide stronger structural protection.

1. **Eliminate through design selection**
2. **Reduce through design alteration**
3. **Add engineered features or devices**
4. **Provide visibility and feedback**
5. **Use procedures, training, equipment**

The best security control may be the attack path the architecture never created.

FUNCTIONAL DESIGN

Make Protection Mechanisms Trustworthy

1. Decision functions
2. Enforcement functions
3. Infrastructure functions
4. Non-bypassable
5. Evaluatable
6. Always invoked
7. Tamper-proof
8. Failure analysis
9. Situational awareness

If a mechanism can be bypassed, disabled at the wrong time, or altered by what it constrains, confidence is fundamentally limited.

ASSURANCE + FRONTIER AI

From Good Design to Justified Confidence

1. Define unacceptable loss
2. Eliminate nonessential capability
3. Separate high-impact domains
4. Constrain AI agent authority
5. Mediate sensitive transactions
6. Analyze protection failure
7. Design trustworthy recovery
8. Verify system-level claims
9. Reassess as adversity changes

AI changes the speed and economics of cyber conflict. It does not repeal the laws of good engineering.

CONCLUSION

Engineer Security from the Inside Out

Trustworthy secure design begins before a security control is selected.

Start with unacceptable loss. Shape the architecture to eliminate or bound it. Add trustworthy mechanisms for what remains. Substantiate the result with evidence.

DESIGN FIRST

Can the loss path or condition be eliminated or structurally reduced?

BOUND COMPROMISE

If compromise occurs, do boundaries still hold and loss remain limited?

SUBSTANTIATE TRUST

Is the evidence strong enough to justify confidence in the design?

Do not ask first which security control to add. Ask why the system permits the unacceptable loss to occur.