



STRATA SECURITY

COMPANY QUALIFICATIONS

Engineering-First Cybersecurity

A publication on how Strata designs, builds, assesses, and sustains secure software systems — for federal technical teams and prime contractors.

BUILT TO SECURE. ENGINEERED TO LAST.

FEDERAL SECURITY SERVICES · 2026 EDITION

Why Strata Exists

An engineering-first company, built on the belief that security is a property you design in — not a report you generate afterward.

Two Gaps, One Practice

Serious application security tooling has rarely been affordable, approachable, or built for teams without dedicated security budgets. At the same time, some software problems — undocumented binaries, legacy systems, mission-critical programs — need direct engineering depth that no automated scan can reach. Strata was built at the intersection of both gaps: a practice that builds real, working security software, and applies that same engineering discipline directly to the programs that need hands-on depth.

Engineering-First, Not Audit-First

An assessment that ends in a list of findings is unfinished work. Strata treats every engagement as an engineering problem: understand the system, identify the root cause of a weakness, and implement or specify a fix that holds up in the actual operating environment — not just in a report.

Secure Software by Design

Security requirements are treated as design constraints from the outset of a system's development, not as a compliance step applied after delivery. This is true of how Strata builds its own Commercial Platform, and it is the same standard applied to every federal and high-assurance engagement.

Mission Statement

Strata Security exists to close the gap between security assessment and security engineering. Mission-focused and high-assurance software programs do not need another audit — they need engineers who can identify a vulnerability, understand its root cause, and implement a fix within the constraints of the operating environment.

Mission & Vision

One engineering practice, two vantage points — each reinforcing the other rather than operating in isolation.

Long-Term Mission

Make practical, high-quality security engineering accessible — to developers and small teams who could never afford enterprise tooling, and to federal programs that need direct engineering depth rather than another vendor layer between them and the work. The goal has not changed since Strata's first line of code: capable security engineering, delivered directly, without unnecessary overhead.

Federal Vision

To be the engineering partner federal technical teams and prime contractors call when a problem needs direct practitioner depth — an undocumented binary, a legacy system, a secure-development program that needs practical implementation support — not a broader advisory relationship layered on top of the work.

Commercial Vision

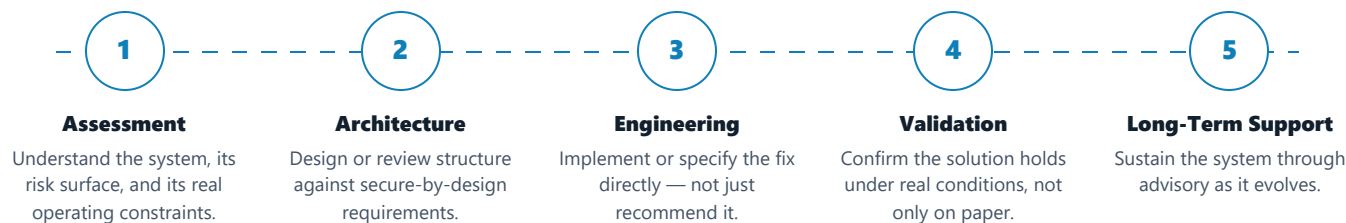
To keep building the Commercial Platform as real, operated software — developer-first, enterprise-inspired but not enterprise-priced, and continuously improved based on how teams actually use it, not on a theoretical roadmap.

How Both Reinforce One Another

The same engineering practice sits behind both. Operating production software surfaces the real constraints — false-positive tradeoffs, remediation guidance that has to actually work, findings that have to be understood by the people who wrote the code — that make federal engineering judgment sharper. In the other direction, the depth required by federal and high-assurance engagements — reverse engineering, vulnerability research, secure SDLC integration — continually raises the bar for what the Commercial Platform is built to detect and explain. Neither practice is a marketing arm for the other; each makes the other's engineering better.

How Strata Approaches an Engagement

Five stages, applied whether the engagement is a fixed-scope assessment or an ongoing advisory relationship.



Assessment Comes First, Always

No engineering recommendation is made before the system, its constraints, and its actual risk surface are understood directly — through code, binaries, and configuration, not documentation alone.

Architecture Is Where Security Gets Cheap or Expensive

Structural decisions made at the architecture stage determine whether security is a design property or a permanent maintenance cost. Strata reviews architecture against the requirements it is meant to satisfy, not against a generic checklist.

Engineering Means Building, Not Just Advising

A finding without an implementable fix is incomplete work. Depending on engagement scope, Strata implements the change directly or specifies it precisely enough that the receiving team can implement it without reinterpretation.

Validation and Long-Term Support Close the Loop

A fix is only real once it has been verified against real conditions, and systems keep evolving after delivery. Many engagements continue as ongoing technical advisory rather than ending at a single deliverable.

Core Technical Capabilities

Eight domains, each grounded in direct engineering practice rather than a generic service-line description.



Secure Software Engineering

Security requirements integrated into design, development, and deployment — treated as a design constraint from the start, not a gate applied at the end.



DevSecOps

Security controls and automated testing embedded directly into CI/CD pipelines, so checks run on every change, not on a schedule.



Mobile Security

Hands-on engineering assessment of mobile application risk — beyond what automated scanning alone can surface.



Reverse Engineering

Analysis of compiled software and firmware to establish behavior and risk when source access is limited or unavailable.



Application Security

Architecture review, threat modeling, and vulnerability assessment across web and application platforms, tied to concrete remediation.



Vulnerability Research

Structured, hands-on research to identify previously unknown vulnerabilities in complex or opaque software.



Cloud Security

Configuration and architecture review for cloud-hosted infrastructure and mission systems.



Defensive Engineering

Design and implementation of controls that reduce attack surface and limit impact when prevention fails.

Secure Development Lifecycle

A closed loop, not a linear checklist — each pass through the cycle feeds the next.



Stage 6 feeds back into Stage 1 — the cycle repeats with each release, not once per program.

Why a Loop, Not a Checklist

A secure development lifecycle that runs once at the start of a program stops reflecting reality the moment the system changes. Strata treats each stage as a recurring discipline: threat models are revisited as the system evolves, architecture is re-reviewed at meaningful design changes, and CI/CD gates apply the same standard to every commit, not just the initial release.

Integration Over Isolation

These six stages are not six separate services. Threat modeling informs what code review looks for; code review findings inform what testing prioritizes; testing results inform what the next architecture review examines. The lifecycle is one connected discipline, applied consistently whether Strata is assessing an existing system or engineering a new one.

Reverse Engineering & Research

A description of method, not a claim of results — specific findings and outcomes are engagement-specific and available on request.

Binary Analysis

Disassembly and decompilation to understand what compiled software actually does, independent of what its documentation claims — necessary whenever source access is limited, unavailable, or not trusted as complete.

Static Analysis

Examining code or binaries without executing them, to identify structural weaknesses — insecure patterns, hardcoded credentials, unsafe primitives — before runtime behavior is even in scope.

Dynamic Analysis

Observing a system's actual running behavior surfaces issues static review alone cannot — timing conditions, runtime configuration, and interactions between components that only appear under execution.

Mobile Application Security

Direct, hands-on assessment of a mobile application's risk surface — storage, communication, binary protections, platform-specific misconfiguration — distinct from, and deeper than, automated scanning.

Research Methodology

A structured, repeatable process: understand intended behavior, form a specific hypothesis about where that behavior could fail, verify it directly against the artifact, and document the finding with reproducible evidence.

Responsible Engineering

Research exists to inform engineering outcomes, not to generate exposure. Findings are documented with reproducible evidence and handled under responsible disclosure practice, scoped to what an engagement authorizes.

What Operating a Production Platform Teaches

Not a product pitch — an explanation of why building real software sharpens federal engineering judgment.

Advising Is Not the Same as Operating

Strata designs, builds, and operates the Commercial Platform, a self-service application security assessment product used directly by developers and small teams. Operating production software is a different discipline than advising on one: it means living with the consequences of every design decision, in front of real users, continuously — not just at the moment a report is delivered.

Real Constraints, Not Theoretical Ones

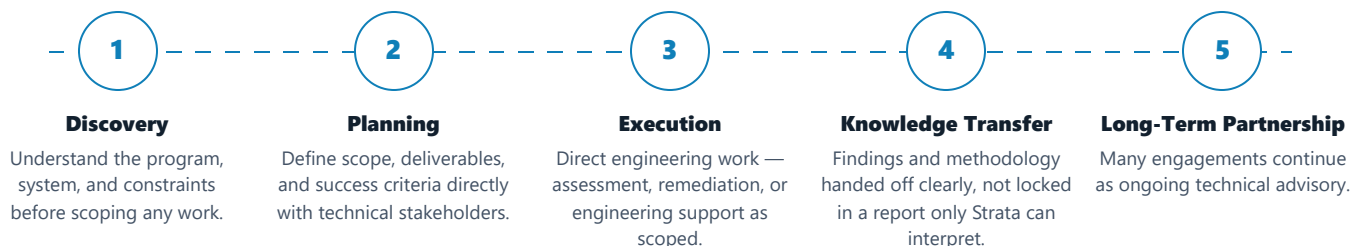
A finding that looks correct in isolation still has to survive contact with a real user: false-positive tradeoffs that matter when someone is triaging results under time pressure, remediation guidance that has to be specific enough to actually implement, and detection logic that has to keep working as the platform itself changes. Building and operating software that has to hold up under those conditions is a standing discipline, not a one-time exercise.

The Feedback Runs Both Ways

Every technical judgment applied in a federal engagement has already been tested against the discipline of shipping and operating working software. In the other direction, the deeper technical work required by federal engagements — reverse engineering, vulnerability research, secure SDLC integration — continually raises the bar for what the platform itself is built to detect, explain, and remediate. The same engineering practice sits behind both.

Federal Engagement Model

Five stages, consistent whether the engagement is a fixed-scope assessment or an ongoing advisory relationship.



Discovery and Planning Are Not Boilerplate

Scope is defined directly with the technical stakeholders who own the system — not assembled from a generic statement of work template. Success criteria are technical and specific, agreed before execution begins.

Execution Is Engineering, Not Just Reporting

Depending on scope, execution means direct remediation work, hands-on assessment, or embedded engineering support — delivered by the same practitioner who scoped the engagement.

The Engagement Doesn't End at the Deliverable

Knowledge transfer means the receiving team understands not just what was found, but why, and how to reason about the next similar issue themselves. Many engagements continue from there as ongoing technical advisory rather than a one-time deliverable.

Why Strata

Five differentiators, each a direct consequence of how Strata is structured — not a claim layered on top of it.

01 **Engineering-First**

Every engagement is treated as an engineering problem to be solved, not an audit to be completed. The standard is a working fix, not a finding.

02 **Founder-Led**

A founder-led technical practice with over a decade of direct experience in reverse engineering, vulnerability research, and technical cybersecurity leadership across high-assurance and government cybersecurity environments.

03 **Direct Practitioner Model**

The engineer who scopes an engagement is the engineer who performs it. No account-management layer sits between the client and the technical work.

04 **Research-Backed Engineering**

Offensive research and defensive engineering are treated as one connected discipline — findings come from direct analysis of real software, not only documentation review.

05 **Long-Term Partnership**

Engagements are structured to continue as ongoing technical advisory where useful, not to maximize the number of separate deliverables sold.

Company Snapshot

Verified facts are stated directly. Registration and certification fields are marked pending — not estimated — until confirmed.

AT A GLANCE

Leadership	Founder-led
Delivery Model	Direct practitioner
Founded	2025
Headquarters	[Pending]
Entity Type	[Pending]
Business Size	[Pending]

CONTACT

Web	strata-security.com/federal
Sales	sales@strata-security.com
General	hello@strata-security.com
Phone	[Pending]
Mailing Address	[Pending]

REGISTRATION & CERTIFICATIONS

UEI	[Pending]
CAGE Code	[Pending]
SAM.gov Status	[Pending]
NAICS	[Pending]
PSC	[Pending]
Certifications	None held

Suggested NAICS to confirm at registration: 541512 (Computer Systems Design Services), 541519 (Other Computer Related Services). PSC intentionally left blank pending confirmation, not guessed.



STRATA SECURITY

strata-security.com/federal

sales@strata-security.com · hello@strata-security.com

BUILT TO SECURE. ENGINEERED TO LAST.