OffSec™
The Path to a Secure Future™

SysteCom
MEMBER OF QNR GROUP

AI: Learning the New Attack Surface

AI systems are becoming critical infrastructure and are already exploitable

Learning What and How to Protect AI Infrastructures

Nikos Niskopoulos

Chief Operations Director at SysteCom

OffSec™
The Path to a Secure Future™

SysteCom
MEMBER OF QNR GROUP

Hello Everyone!

Nikos Niskopoulos

COO | CISO | DPO

MSc OSCP

Every Technology Shift Expands the Attack Surface

Each major transition in computing has redefined how systems are attacked not just where they are exposed.

1

Networks

Expanded connectivity enabled remote access and lateral movement across systems.

2

Web Applications

Business logic became externally accessible, introducing input-driven exploitation.

3

Cloud

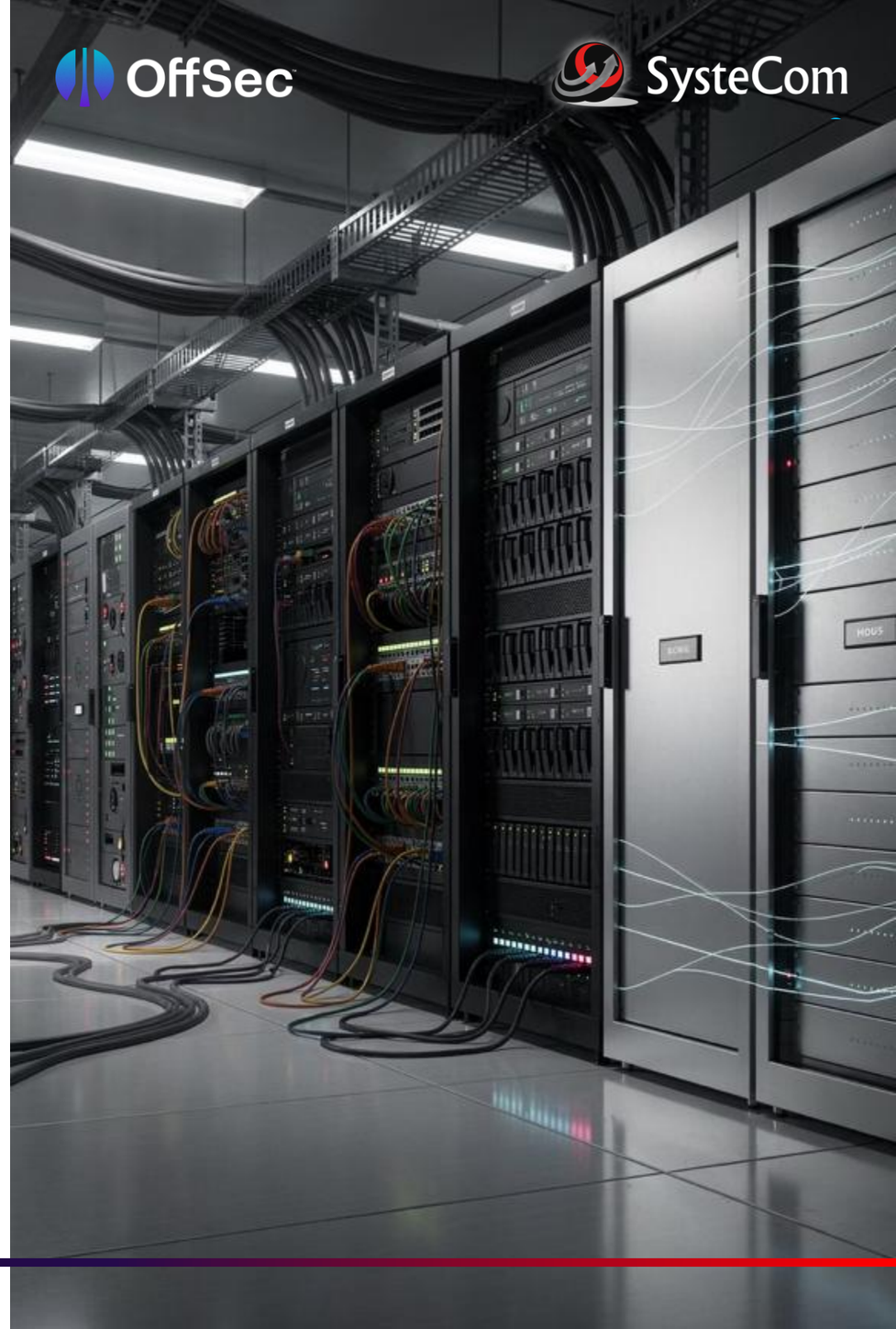
Control shifted to identity and configuration, increasing complexity and scale of misconfigurations.

4

AI

Systems no longer just execute logic they interpret data and make decisions, creating new ways to influence outcomes.

AI changes not just where systems **are exposed**, **but how their behavior can be influenced.**



AI Is Becoming Part of Decision-Making

Artificial intelligence is no longer a standalone tool. It is becoming embedded into how organizations and government entities interpret information, make decisions, and act in real time.

Interpret & Analyze Data

AI processes and interprets data streams to extract meaning and identify patterns.

Drive Decisions

Outputs inform or directly influence decisions across critical functions.

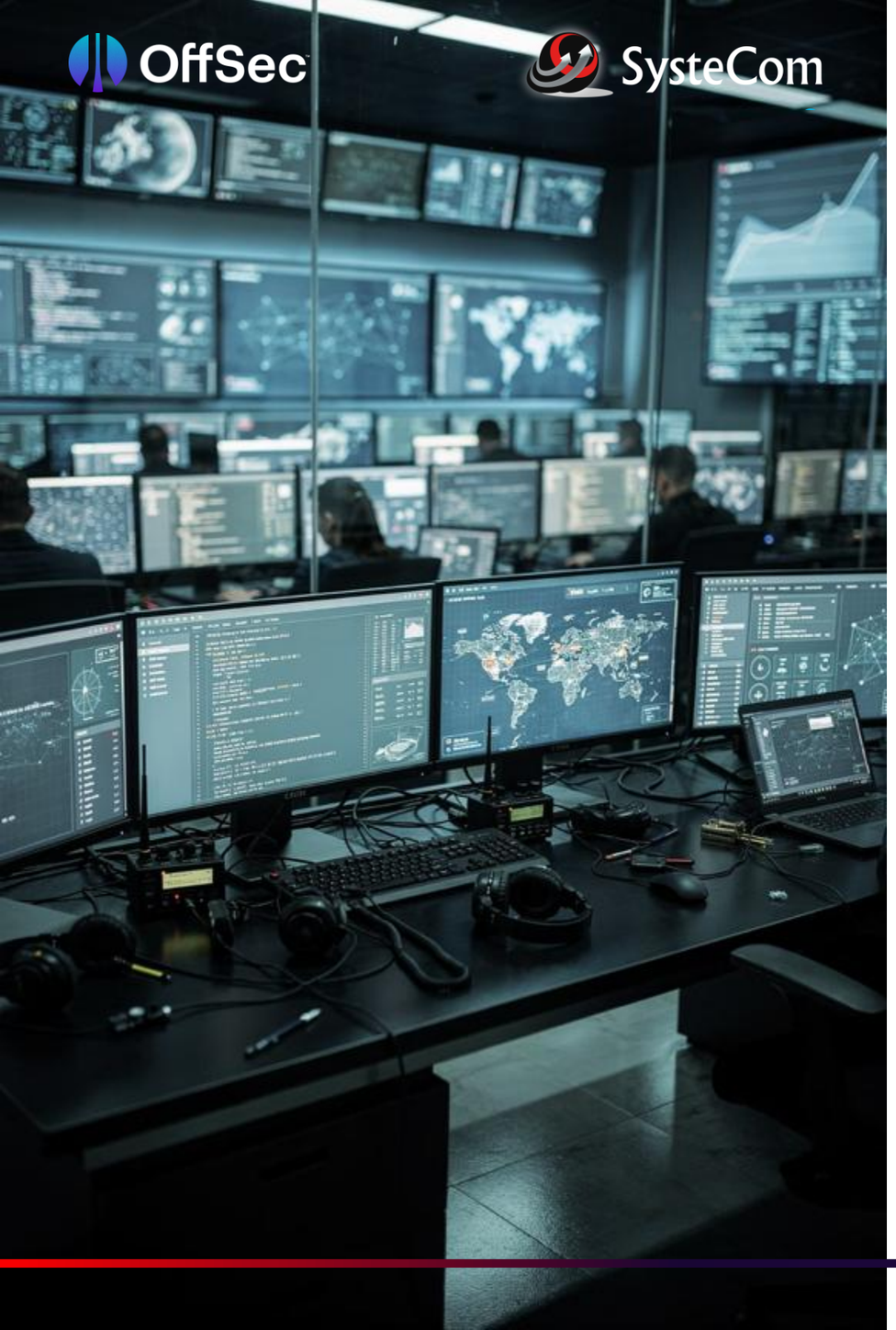
Mediate Interaction

AI systems sit between users, data, and other systems shaping how inputs are handled and responses are generated.

Trigger Downstream Actions

Outputs initiate actions such as API calls, system changes, or automated workflows.

These systems are not passive — their **outputs influence real outcomes**.



How AI Is Being Used in Enterprise Government & Defense

AI is becoming a core capability across enterprises government and defence systems, and its adoption is **no longer optional**.

Movement Detection & Tracking

AI is used to detect and track vehicles, aircraft, or vessels from satellite and drone imagery.

Cybersecurity Operations

AI is used to identify unusual login patterns, lateral movement, or unexpected data transfers across networks.

Monitoring Restricted Areas

AI is used to monitor video, radar, and sensor feeds and alert on movement or presence in controlled zones.

Logistics & Resource Prediction

AI is used to predict resource needs, identify shortages, and highlight potential disruptions in logistics and resupply.

Intelligence Priorization

AI is used to condense large volumes of intelligence reports into prioritized summaries for analysts.

On the surface, these systems look different — but underneath, they share a typical, common **architecture.**

Modern AI Systems Are Composed of Layers

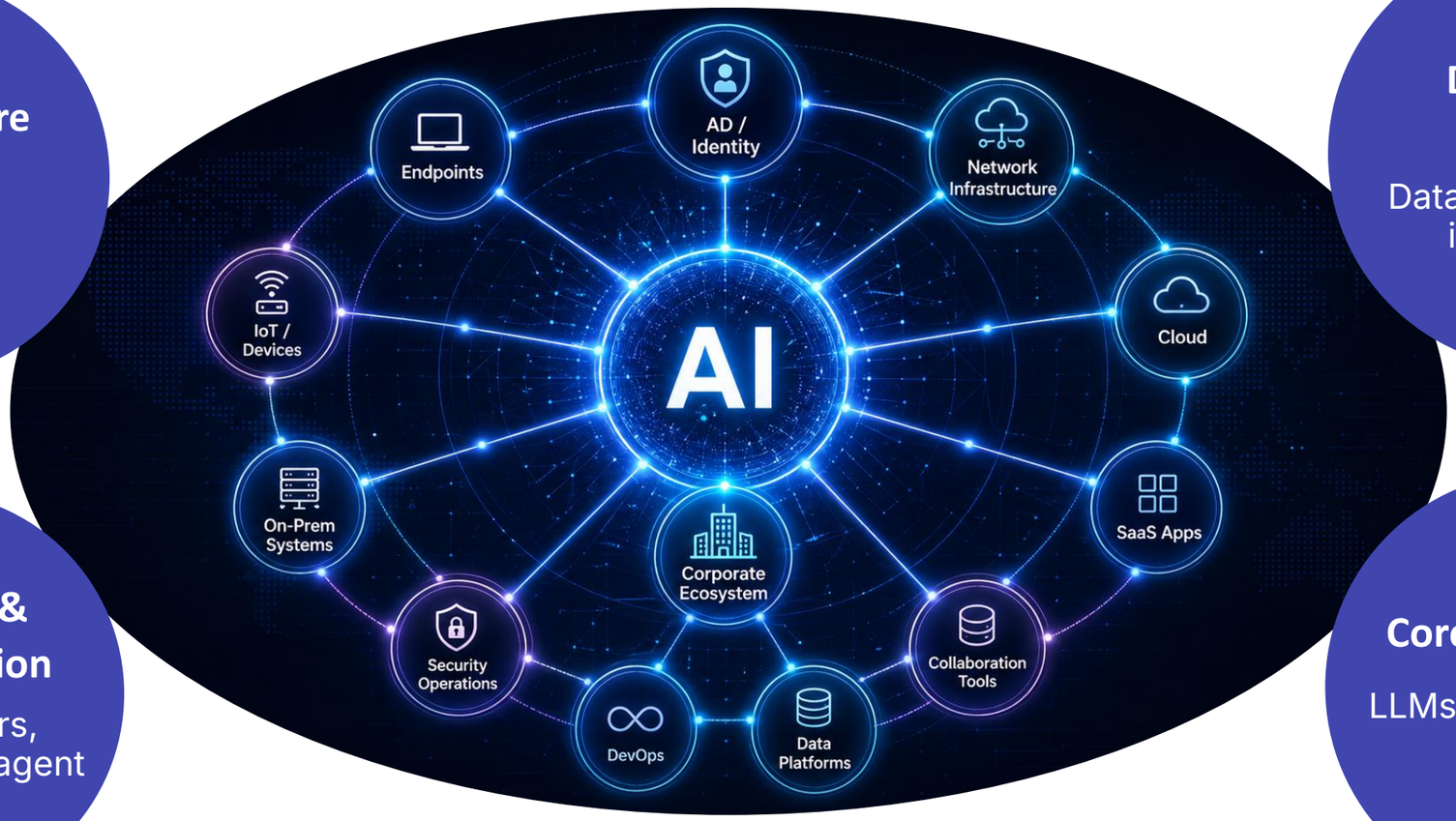
A production AI system is not a single model. It is a pipeline of interdependent components — each introducing its own trust assumptions, failure modes, and exploitable interfaces.

Infrastructure

Cloud
On prem

Retrieval & Orchestration

RAG, vectors, prompts, and agent logic



Data, Tools & Outputs

Data Sources, APIs, integrations, and actions

Core Intelligence

LLMs and ML models powering reasoning

This is a **chain of dependencies**, not a single system. Each layer extends the attack surface.

The Shift: From Logic to Behaviour

Traditional security assumes systems behave predictably and can be controlled through code and boundaries. AI breaks these assumptions.

Traditional Security Focus

- Controlling logic and execution paths
- Enforcing access and system boundaries
- Verifying behaviour through testing and rules

AI Security Focus

- Securing data sources and trustworthiness
- Managing emergent and unpredictable behaviour
- Controlling interactions at runtime

AI systems **inherit risk from the data they consume and express that risk through their behaviour.**

The Attack Surface Is Distributed

Every component in an AI pipeline rests on implicit trust assumptions. Individually, each assumption seems reasonable. Collectively, they form a fragile chain and chains fail at their weakest link.

Data is Trusted

Training and retrieval sources are assumed to contain clean, unmanipulated content.

Models Behave as Expected

The model produces outputs consistent with its stated design and guardrails.

Retrieved Content Is Safe

Documents injected via RAG are assumed benign and contextually appropriate.

Prompts Are Followed Faithfully

System instructions are assumed to be honoured throughout the interaction.

Tools Are Used Correctly

External integrations are invoked only as intended, with appropriate scope.

Interconnections are Secure

Interconnected systems, environments, interfaces interact as expected.

If **any assumption fails**, the system can be influenced silently and without a traditional exploit.

Mapping Threats Across the Architecture

AI threats do not map neatly to traditional vulnerability classes. Instead, they correspond to specific components in the AI pipeline each with distinct manipulation techniques and impact paths.

Component	Threat	Impact
Data / RAG	Poisoned or malicious content injection	Corrupted context, biased outputs
Model Layer	Backdoors, hidden or latent behaviour	Triggered misclassification or bypass
Orchestration	Prompt injection, instruction override	Safeguard bypass, privilege escalation
Tools & APIs	Unauthorised or excessive tool invocation	Data exfiltration, unintended actions
Outputs	Misleading, unsafe, or weaponised responses	User deception, downstream harm

Attacks **influence** the system rather than break it — producing wrong outcomes through legitimate mechanisms.

Runtime Manipulation Is the Key Risk

The majority of AI attacks do not require access to model weights, training pipelines, or source code. They occur during normal usage through the system's own intended interface.

Crafted Inputs

Adversarially constructed inputs shift model outputs without triggering anomaly detection.

Prompt Injection

Embedded instructions override system-level guardrails and redirect model behaviour.

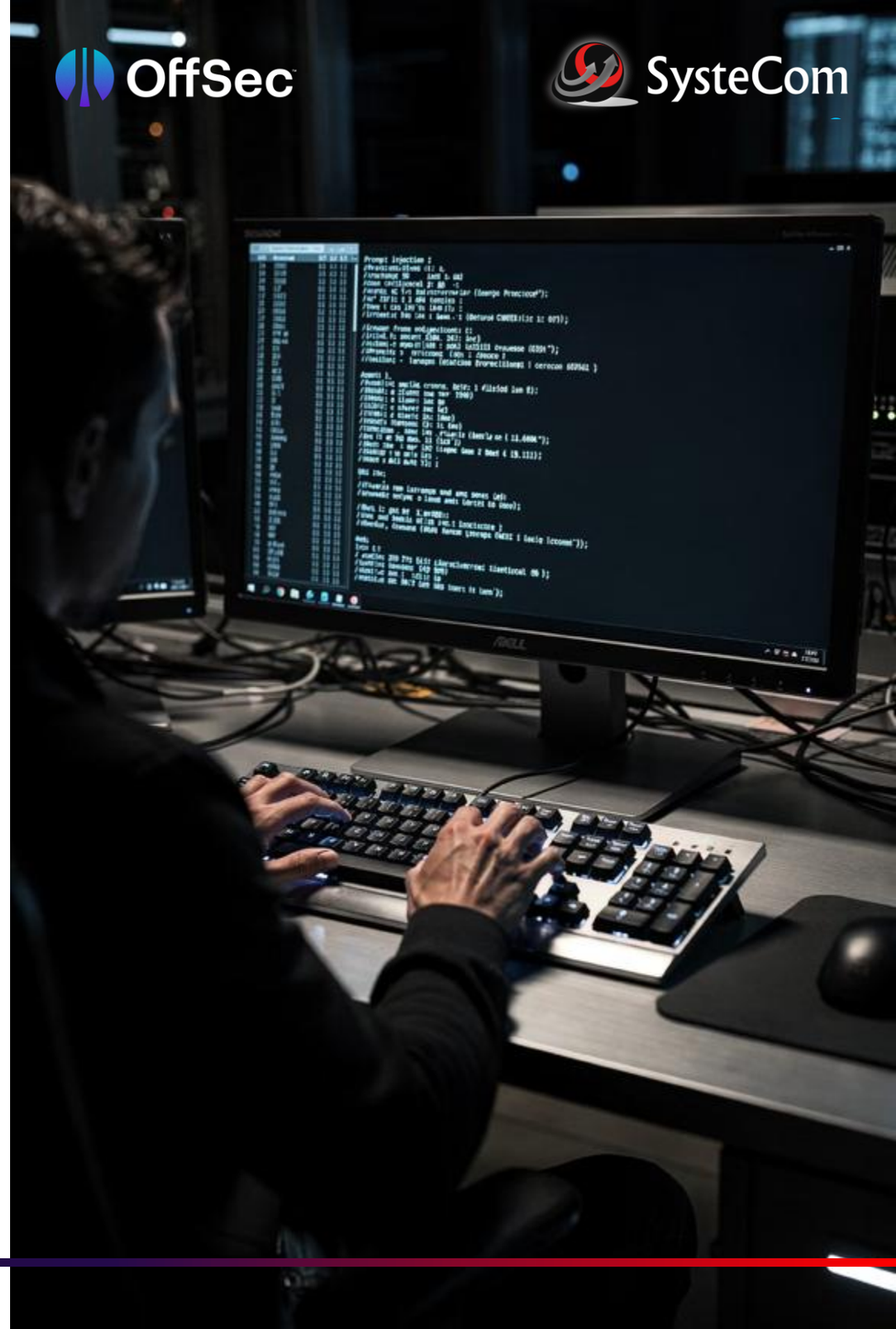
RAG Poisoning

Malicious content introduced via retrieval provides attacker-controlled context to the model.

Tool Manipulation

Model reasoning is steered to invoke external tools in unintended or unauthorised ways.

The system behaves **as designed** — but produces the wrong outcome.



Where is the AI Defense GAP?

AI threats are structurally resistant to conventional defence strategies. The properties that make these systems hard to attack in the traditional sense also make them hard to protect. **We need to shift from Linear to Graph...**

Malicious Inputs Look Normal

Adversarial prompts are syntactically valid. Signature-based detection is ineffective there is nothing anomalous to flag.

Behaviour Is Context-Dependent

The same input in a different context may produce a different, harmful output. Consistency cannot be assumed.

Trust is Distributed

Responsibility for safety is spread across data owners, model developers, and system integrators with no single accountable party.

Testing Requires a Different Approach

Testing must address the full system not just the model, but the data, orchestration, the integration it depends on as well as the runtime behaviour.

All of this leads to one conclusion — **AI systems must undergo extensive security testing before production, but this testing must take a different approach, focused on behaviour, interaction, and the systems and data around the model.**

What AI Security Testing Should Look Like

AI penetration testing is a discipline distinct from traditional application security assessment. The objective is not to find a code vulnerability it is to demonstrate that the system can be made to produce harmful, unintended, or policy-violating outcomes through normal interaction.

01

Map the AI Attack Surface

Identify where and how AI is integrated: models, agents, RAG pipelines, tool layers, and external dependencies

02

Adversarial Behavioural Testing

Test how the system responds to crafted, boundary-pushing, and indirect inputs.

03

RAG Pipeline Exploitation

Assess how external knowledge sources can be poisoned, manipulated, or hijacked.

04

Agent and Tool Abuse

Manipulate agent logic and tool invocation to trigger unintended or privileged actions

05

Assess Interaction & Control Boundaries

Attempt prompt injection, system instruction bypass, and multi-turn manipulation sequences.

Testing must account for both **runtime interaction** risks and **upstream supply chain weaknesses**.

AI Is Not Just a Capability.

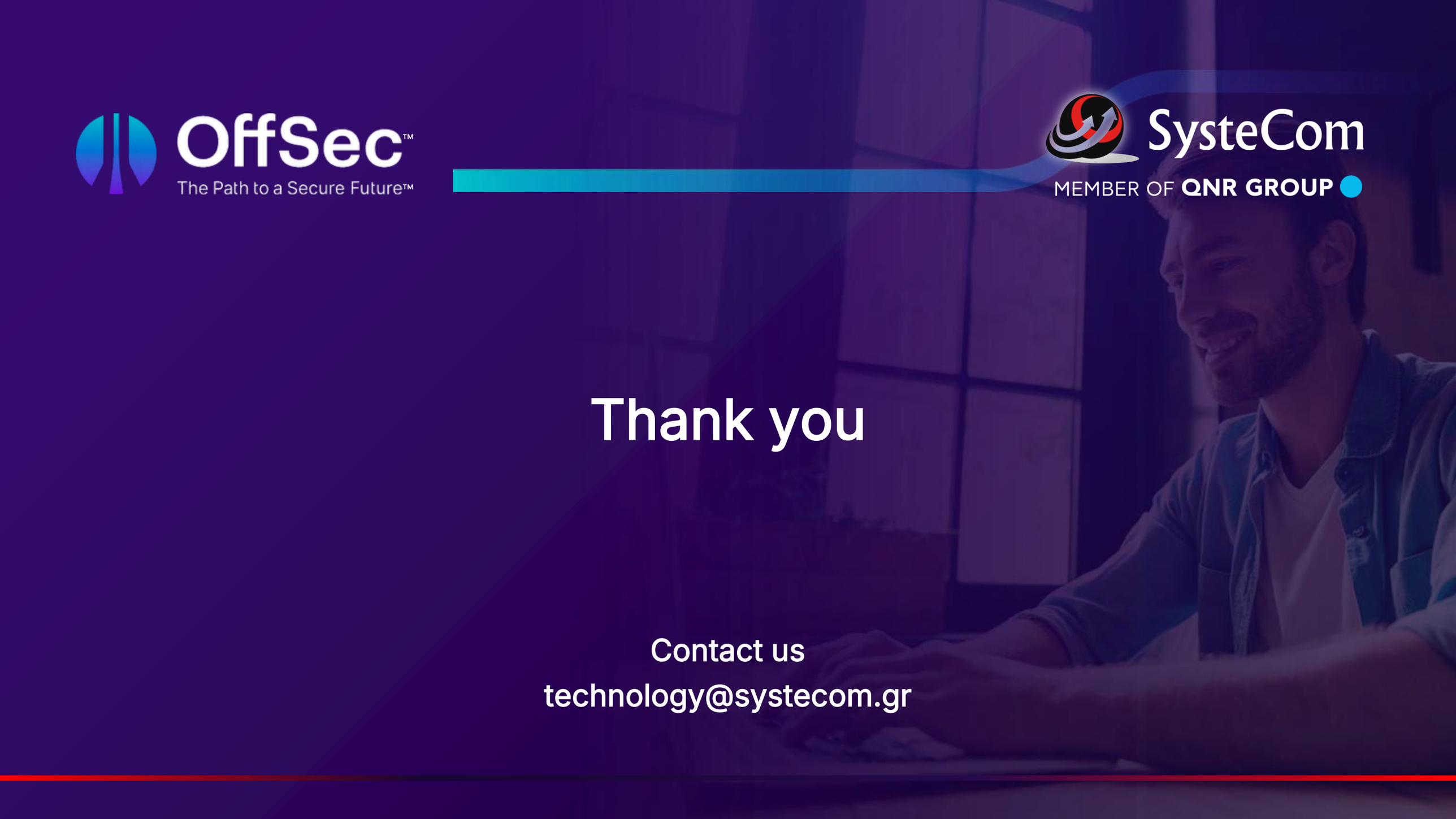
The Attack Surface Expands & The Potential Impact Gets Infinite...

As AI systems are embedded into enterprises, critical infrastructure, defence platforms, and decision-making pipelines, the traditional model of security defined by intrusion, access, and code exploitation is no longer sufficient.

The defining characteristic of AI risk is influence, not intrusion. Understanding this distinction is the first step in securing the systems that will increasingly shape consequential outcomes.

OffSec™
The Path to a Secure Future™

SysteCom
MEMBER OF QNR GROUP



Thank you

Contact us
technology@systecom.gr