HORIZON3

# From a Cracked CEO Credential to a Potential Ransomware Path

A controlled NodeZero test cracked one CEO credential and mapped three potential downstream impacts, prompting a password change and focused remediation.

CASE STUDY

## Customer *snapshot*

**CUSTOMER**

Anonymous gaming and hospitality operator

**INDUSTRY**

Gaming, hospitality, and entertainment

**ENVIRONMENT**

Multi-property casino, hotel, and entertainment operations, with network segmentation and endpoint detection

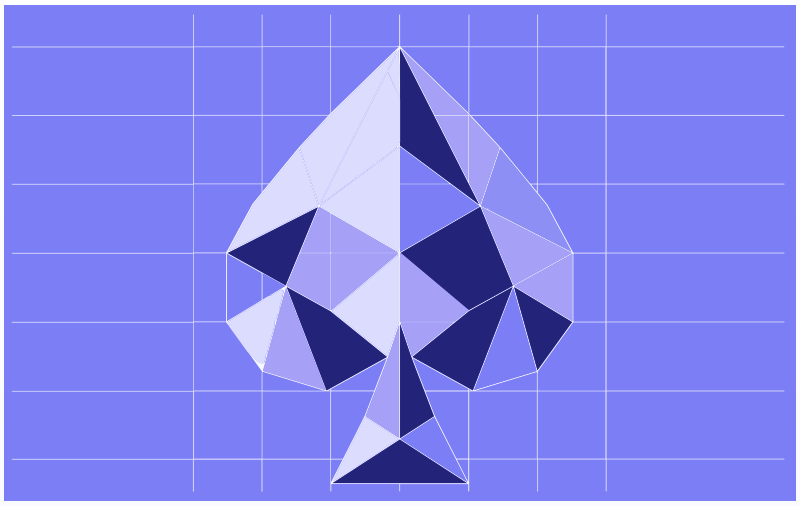
**CHAMPION**

Security leader

**SOLUTION**

Horizon3 NodeZero Internal Pentest, simulating a compromised executive workstation and tracing the path from NBT-NS poisoning to credential compromise

**CHALLENGE**



### The organization operates casino, hotel, and entertainment services across multiple properties, with network segmentation and endpoint detection among its existing defenses

Its security team had set a phased validation goal: start with broad coverage, address high-priority findings, then test from the perspective of a compromised workstation inside a segmented network. The next question was more focused: if an attacker gained a foothold in an executive segment, what could that access lead to, and which control would break the chain?

WHY NOW?

## A chain from interception to an usable credential

The team used NodeZero to simulate a compromised executive workstation. The first proof was concrete. NodeZero used NBT-NS poisoning, a name-resolution weakness that can expose authentication traffic, to capture an NTLMv2 hash, a protected representation of a password exchanged during authentication. It then cracked the hash and recovered a usable CEO credential. When the security leader asked how the hash could be captured in a switched network, the attack-path evidence showed each step. “It intercepted and then cracked, the CEO’s password.” The CEO had initially questioned whether the credential could really be recovered; the evidence changed the conversation.

### MFA stopped the first attempts, then a token path surfaced

A second moment showed both the value and the limits of MFA. The initial authentication attempts triggered MFA and were blocked, alerting the CEO to the activity. The exercise also surfaced a token issued after a quick approval in the password-change flow. The lesson was specific: MFA interrupted the first attempts, while the credential and token-handling path still needed attention.

### From identity compromise to potential business exposure

The third proof connected identity to business exposure. NodeZero mapped potential paths from the compromised account toward domain-user compromise, ransomware exposure, and sensitive-data exposure. These were demonstrated attack paths in a controlled exercise, not a real ransomware event or data breach.

WHY HORIZON3?

## Evidence that connected weakness to consequence

For this team, NodeZero’s value was the evidence connecting cause to consequence. It answered how the credential had been intercepted, showed which authentication attempts MFA stopped, and distinguished the proven credential compromise from the downstream outcomes that remained potential. That gave security leadership a practical basis for deciding what to change, rather than relying on a severity label alone.

VERIFIABLE RESULTS

## Immediate actions

**The evidence led to immediate, focused action:**

- Executive response: The CEO changed the password after seeing the proof.
- Remediation priorities: The team prioritized the NBT-NS interception path and raised unnecessary SMBv1 use for broader review.
- Control clarity: The exercise confirmed that MFA blocked the initial attempts, while identifying a separate token-related path for review.

### A clearer basis for verification

The team could now explain the exposure as a sequence of testable control decisions: where authentication traffic was intercepted, what stopped the first login attempts, and which changes could interrupt the path.

## Looking Ahead

The next opportunity is to retest after the prioritized changes, then move the starting point to another segment to learn what a different compromised endpoint could reach. The program now has a forward test: prove whether each control breaks the path before an attacker can use it.