Vantalect



AWS INCIDENT RESPONSE RETAINER · SAMPLE DELIVERABLE

Activation Role and Drill

DISTRIBUTION: OPERATIONAL

Week 1 of the Retainer: Access · prepared for Wrenfield Bancorp, Inc.

Prepared by Charles Sieg · charles.sieg@vantalect.com · vantalect.com

August 12, 2026 · AWS Incident Response Retainer (one year)

SAMPLE DELIVERABLE

This is a sample deliverable. Wrenfield is fictional. Its people, customers, accounts, figures, and attacker are invented for this sample, and any resemblance to a real bank, company, person, or threat group is coincidental. The attacker's addresses come from ranges reserved for documentation.

Contents

What this report is **2**

The role's design **3**

Who may assume it 3

What it may do, and what it may never do 4

The hold 7

The StackSet **8**

The drill **9**

Results by account **11**

The one finding 12

The management account 13

The one-call activation **13**

Compared with July 4 ..... **14**

What is open **14**

Notice **15**

What this report is

On Wednesday, August 12, 2026, the activation role worked in all 22 accounts, and the first read came 9 minutes after the activation call. Mireille Danvers, this is the record of the drill: how the role is built, how it was deployed, what we did minute by minute, and what we found. One thing failed the first time (the vault account), and it was fixed the same morning.

This report is for the bank's incident and platform teams. It is not part of the counsel track and holds no legal analysis. The report covers the activation role and its drill. It does not cover the log-readiness check (it starts Monday, August 17, 2026), Interceptor's watch (scheduled for Wednesday, August 19, 2026), or the tabletop (Tuesday, September 15, 2026). It holds no legal analysis. The drill was a rehearsal: no incident was open, no real evidence was collected, and no containment was proposed.

The drill in six measures.

Measure	Result
Activation call to first read	9 minutes (target in the Retainer: under 15 minutes)
Accounts where the role read, after activation	22 of 22 (21 on the first pass, 22 after one fix)
Accounts where a write was denied	22 of 22; in the evidence account the role may write to one bucket only
Accounts that denied every call after the hold returned	22 of 22
Calls the approver made to activate	One
Findings	One (D-1), fixed during the drill

The role's design

The role can read, can write evidence to one bucket in one account, and can propose containment on paper. It cannot change anything else, and it can do nothing at all until the bank's approver removes the hold. Four pieces make that true: who may assume it, what it may do, what it may never do, and the hold.

Who may assume it

The role `VantalectIncidentResponse` exists in every account of the organization. Its trust policy admits one named principal, the IAM user `charles-sieg` in Vantalect's account `111122223333`, not the account as a whole, and only when three conditions hold: the sign-in carried multi-factor authentication, the caller supplied the external ID the bank holds, and the caller set the source identity `charles.sieg`, which every call in the session then carries into the trail. The multi-factor context comes from a `GetSessionToken` session that Charles opens for that IAM user with his hardware key; an Identity Center session does not carry it. The external ID below is an example value for this sample. The role the bank deployed on July 4 already named the same one user with the same multi-factor and source-identity conditions; the Retainer role keeps that and adds the external ID. It lists `sts:SetSourceIdentity` beside `sts:AssumeRole`, because AWS denies an assume-role call that sets a source identity unless the trust policy allows that action.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VantalectNamedPrincipalOnly",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:user/charles-sieg"
      },
      "Action": [
        "sts:AssumeRole",
        "sts:SetSourceIdentity"
      ],
      "Condition": {
        "Bool": {
          "aws:MultiFactorAuthPresent": "true"
        },
        "StringEquals": {
          "sts:ExternalId": "wf-ir-2026-8f3a1c",
          "sts:SourceIdentity": "charles.sieg"
        }
      }
    }
  ]
}
```

A session lasts at most 4 hours. A second responder cannot be added without the bank editing this policy, which is the intended friction: the Retainer has one responder, and the paper says so.

What it may do, and what it may never do

The permissions policy has three allow statements and one explicit deny. The allow statements cover reading (the audit and log services, identity, network, storage, database, and backup metadata), writing evidence to the bucket `wf-ir-evidence` in the evidence account `584729107701` (`wf-ir-forensics`), and checking a containment policy before the bank attaches it. The deny blocks every write family on any resource except the evidence bucket and its key. The listing below is shortened to the action families; the deployed template carries the full lists.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ReadForInvestigation",
      "Effect": "Allow",
      "Action": [
        "cloudtrail:Describe*",
        "cloudtrail:Get*",
        "cloudtrail:List*",
```

```
        "cloudtrail:LookupEvents",
        "cloudtrail:StartQuery",
        "cloudtrail:GetQueryResults",
        "config:Describe*",
        "config:Get*",
        "config:List*",
        "guardduty:Get*",
        "guardduty:List*",
        "iam:Get*",
        "iam:List*",
        "kms:Describe*",
        "kms:Get*",
        "kms:List*",
        "organizations:Describe*",
        "organizations:List*",
        "route53:Get*",
        "route53:List*",
        "glue:Get*",
        "ec2:Describe*",
        "rds:Describe*",
        "backup:Describe*",
        "backup:List*",
        "s3:GetBucket*",
        "s3:ListAllMyBuckets",
        "sso:List*",
        "sso:Describe*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "WriteEvidenceOnly",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:AbortMultipartUpload",
        "s3:ListBucket",
        "kms:GenerateDataKey",
        "kms:Encrypt",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:s3:::wf-ir-evidence",
        "arn:aws:s3:::wf-ir-evidence/*",
        "arn:aws:kms:us-east-1:584729107701:key/6804741d-d75a-521f-90c0-7cb2e10c3ad2"
      ],
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "s3.us-east-1.amazonaws.com"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:s3:arn": [
            "arn:aws:s3:::wf-ir-evidence",
            "arn:aws:s3:::wf-ir-evidence/*"
          ]
        }
      }
    },
```

```
    {
      "Sid": "ProposeContainmentOnPaper",
      "Effect": "Allow",
      "Action": [
        "access-analyzer:ValidatePolicy",
        "iam:SimulateCustomPolicy",
        "iam:SimulatePrincipalPolicy"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyEveryWriteExceptEvidence",
      "Effect": "Deny",
      "Action": [
        "s3:Put*",
        "s3:Delete*",
        "s3:Create*",
        "s3:Replicate*",
        "kms:Create*",
        "kms:Put*",
        "kms:Import*",
        "kms:Delete*",
        "kms:Disable*",
        "kms:Schedule*",
        "iam:Create*",
        "iam:Put*",
        "iam:Delete*",
        "iam:Attach*",
        "iam:Detach*",
        "iam:Update*",
        "ec2:Create*",
        "ec2:Delete*",
        "ec2:Modify*",
        "ec2:Terminate*",
        "ec2:Stop*",
        "rds:Create*",
        "rds:Delete*",
        "rds:Modify*",
        "organizations:Attach*",
        "organizations:Detach*",
        "organizations:Create*",
        "organizations:Delete*",
        "organizations:Update*",
        "cloudtrail:Stop*",
        "cloudtrail:Delete*",
        "cloudtrail:Update*",
        "cloudtrail:Put*",
        "backup:Delete*",
        "backup:Put*"
      ],
      "NotResource": [
        "arn:aws:s3:::wf-ir-evidence",
        "arn:aws:s3:::wf-ir-evidence/*",
        "arn:aws:kms:us-east-1:584729107701:key/*"
      ]
    }
```

```
    ]
  }
```

The role cannot read stored data, attach a policy, stop a trail, or delete a snapshot. It holds `kms:Decrypt` only on the evidence key, only through S3, and only for the evidence bucket's encryption context, because an upload to a bucket encrypted with a KMS key (a multipart upload most of all) needs `Decrypt` as well as `GenerateDataKey`. That permits the encryption step of an upload and grants no `s3:GetObject`, so it reads no object. The drill tested a complete multipart upload. Containment is the bank's act: we propose, a named approver of the bank approves, and the bank's platform lead attaches. That was the order on July 4, and the role now makes it the rule rather than the habit.

The evidence account needed its own policy change, and the bank made it. The July and hand-off custody logs record that the evidence account's deny-all policy, `wf-evidence-deny-all`, exempts only the bank's custodian role, `wf-evidence-custodian`, and that Vantalect held no role there. The StackSet puts `VantalectIncidentResponse` in the evidence account like every other account, but under that policy it could do nothing. On Tuesday, August 11, 2026 at 15:45 EDT, Mireille Danvers approved in writing and Imani Castell, as custodian, changed `wf-evidence-deny-all` to also exempt `VantalectIncidentResponse` for actions on the evidence bucket `wf-ir-evidence` and its key only, and added the role to the key policy for `Encrypt`, `GenerateDataKey`, and `Decrypt` through S3 only. The change gives the role no way to delete an object, change a retention or a hold, or alter any policy, and the Object Lock default still applies to what it writes. The read-and-upload test is the drill's evidence write on Wednesday, August 12, 2026. The change is after the custody logs were closed on August 7 and is recorded in the bank's change record; the earlier statements that Vantalect held no role in the evidence account were true when they were written.

The hold

Until the approver removes it, a client-owned deny policy stops every call the role makes. The hold is a service control policy attached to the organization root, owned and stored by the bank. Vantalect cannot remove it: the role's own deny statement above blocks `organizations:DetachPolicy`, and the approver's role belongs to the bank.

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "HoldVantalectIncidentRole",
      "Effect": "Deny",
      "Action": "*",
      "Resource": "*",
      "Condition": {
        "ArnLike": {
          "aws:PrincipalArn": "arn:aws:iam::*:role/VantalectIncidentResponse"
        }
      }
    }
  ]
}
```

The hold is a deny that the bank owns and attaches. It stops what the role can do, not the act of assuming it: before activation the role can be assumed, and every call returns an explicit deny until the bank's approver removes the hold. We chose that on purpose: it lets us prove the deny works before an incident, in the same sign-in we would use during one. A service control policy does not apply to the management account, so there the hold is the same deny attached to the role as an identity policy. The activation call removes both.

The StackSet

The role went to the organization on Tuesday, August 11, 2026 as one StackSet, in 21 accounts by the organization's automatic deployment and in the management account by a single stack from the same template. The management account is separate because StackSets with service-managed permissions do not deploy there. The template is Vantalect's, the bank read it before deploying, and the bank's platform lead, Imani Castell, ran the deployment.

The StackSet and what it deployed.

Item	Value
Template	`VantalectIncidentResponse.yaml`, the file the bank deployed on July 4 with the hold and the evidence-only write added
Permissions model	Service-managed, target: organization root, Regions: us-east-1
Instances	21 in the StackSet, 1 stack in `wf-management`
Accounts	22 (20 before July 4, plus the evidence account and the recovery vault)
Deployed by	Imani Castell
Drift check, August 12 before the drill	No drift in any instance
Role, trust, and permissions	As listed above; identical in every account

On July 4 the same role went out to the 20 accounts then in the organization within a minute: a self-managed StackSet created it in 19 accounts from 01:39:51 EDT to 01:40:09 EDT, through the `wf-break-glass` execution role that the containment policy exempted, and a separate stack created it in `wf-management` at 01:40:31 EDT. The minutes that mattered were the ones before the StackSet was created at 01:31:40 EDT. The last section compares the two days.

The drill

The clock ran from counsel's call to the first successful read. It started at 10:00 EDT and stopped at 10:09 EDT: 9 minutes. The Retainer's commitment for this test is under 15 minutes. Counsel's pre-agreed statement of work was signed in the drill by Diane Whitcombe; the bank's approver made one call.

The drill, minute by minute. Times are Eastern Daylight Time, Wednesday, August 12, 2026.

Time (EDT)	Who	What happened	Evidence
10:00 EDT	Diane Whitcombe	Diane Whitcombe calls Charles on the incident line and says the drill word. The activation clock starts.	Drill log, entry 1
10:02 EDT	Diane Whitcombe	Diane emails the signed drill copy of the pre-agreed incident statement of work to Mireille Danvers and Charles.	Email, 10:02 EDT
10:04 EDT	Mireille Danvers	Mireille runs the one activation call from the management account.	`DetachPolicy` and `DetachRolePolicy` in the trail
10:05 EDT	Mireille Danvers	The call returns: the check finds the hold detached in all 22 accounts.	Script output
10:07 EDT	Charles Sieg	Charles opens an MFA session for his IAM user with the hardware key (`GetSessionToken`) and assumes the role in wf-log-archive with the external ID and source identity.	`AssumeRole` in the trail
10:09 EDT	Charles Sieg	First read: `cloudtrail:DescribeTrails` in wf-log-archive returns the organization trail.	Trail event in wf-log-archive
10:31 EDT	Charles Sieg	The first sweep ends: 21 of 22 accounts read. wf-recovery-vault refuses the role (see below).	Sweep log
10:50 EDT	Imani Castell	Imani Castell adds a read-only exception for the role to the vault account policy and attaches the revised policy.	Change record, wf-management
10:58 EDT	Charles Sieg	wf-recovery-vault retested: the read succeeds, a write is denied. 22 of 22.	Sweep log
11:05 EDT	Mireille Danvers	Mireille runs the hold call: the deny is attached again in all 22 accounts.	Script output
11:12 EDT	Charles Sieg	Retest after the hold returns: all 22 accounts deny every call from the role. The drill ends.	Sweep log

Where the 9 minutes went: 2 for counsel to call and sign, 3 for the approver to run the call, and 4 on our side to sign in, assume the role, and read. Our side is the part that does not change with the size of the bank.

Results by account

The role read in all 22 accounts, wrote nowhere except the evidence bucket, and was shut out again when the hold returned. The first read in each account used a harmless describe or list call chosen for that account's purpose. The write test was `iam:CreateRole` for a role named `drill-should-fail`, which every account denied. In the evidence account we also uploaded one test object to the bucket in several parts under the bucket's KMS key (a complete multipart upload), which succeeded after the bank's policy change described next. That object went to a `drill/` prefix, and the custody log records it as a drill artifact and not evidence. A bank that prefers can point the next drill at a separate test bucket.

Per-account results of the drill. The vault account read only after the fix described below.

Account	Account ID	First read	Read at (EDT)	Write test	After hold
wf-management	`584729103301`	`organizations:DescribeOrganization`	10:12 EDT	Denied	Denied
wf-log-archive	`584729103314`	`cloudtrail:DescribeTrails`	10:09 EDT	Denied	Denied
wf-audit	`584729103322`	`guardduty:ListFindings`	10:11 EDT	Denied	Denied
wf-network	`584729103337`	`ec2:DescribeTransitGateways`	10:14 EDT	Denied	Denied
wf-shared-services	`584729103345`	`route53:ListHostedZones`	10:15 EDT	Denied	Denied
wf-identity	`584729103351`	`sso:ListInstances`	10:16 EDT	Denied	Denied
wf-tools	`584729103358`	`iam:ListUsers`	10:17 EDT	Denied	Denied
wf-backup	`584729103366`	`backup:ListBackupVaults`	10:18 EDT	Denied	Denied
wf-prod-lending	`584729104402`	`rds:DescribeDBSnapshots`	10:19 EDT	Denied	Denied
wf-prod-imaging	`584729104415`	`s3:GetBucketVersioning`	10:20 EDT	Denied	Denied

Account	Account ID	First read	Read at (EDT)	Write test	After hold
wf-prod-data	`584729104423`	`glue:GetDatabases`	10:21 EDT	Denied	Denied
wf-prod-digital	`584729104438`	`ec2:DescribeInstances`	10:22 EDT	Denied	Denied
wf-prod-treasury	`584729104446`	`ec2:DescribeInstances`	10:23 EDT	Denied	Denied
wf-nonprod-lending	`584729105502`	`rds:DescribeDBInstances`	10:24 EDT	Denied	Denied
wf-nonprod-imaging	`584729105515`	`s3:GetBucketVersioning`	10:25 EDT	Denied	Denied
wf-nonprod-data	`584729105523`	`glue:GetDatabases`	10:26 EDT	Denied	Denied
wf-nonprod-digital	`584729105538`	`ec2:DescribeInstances`	10:27 EDT	Denied	Denied
wf-nonprod-treasury	`584729105546`	`ec2:DescribeInstances`	10:28 EDT	Denied	Denied
wf-sandbox-1	`584729106601`	`ec2:DescribeInstances`	10:29 EDT	Denied	Denied
wf-sandbox-2	`584729106615`	`ec2:DescribeInstances`	10:30 EDT	Denied	Denied
wf-ir-forensics	`584729107701`	`s3:ListBucket (wf-ir-evidence)`	10:31 EDT	Evidence write allowed (a complete multipart upload, after the August 11 policy change in the evidence account); `iam:CreateRole` denied	Denied
wf-recovery-vault	`584729107714`	`backup:ListBackupVaults`	10:58 EDT	Denied	Denied

The one finding

D-1: the vault account refused the role. The vault account sits behind a deny-all policy with two exceptions, the copy roles, so the incident role was refused even after the hold came off. The sweep logged an explicit deny from a service control policy at 10:31 EDT. The fix was: a read-only

exception for the role, by principal name, added to the vault account policy. Writes stay denied. Imani Castell made the change at 10:50 EDT, and the retest at 10:58 EDT read the vault metadata and was denied a write. The bank's other guardrails were not touched.

The finding matters for the real thing. On July 4 the vault account did not exist when the role went out, and a drill is the cheap time to learn that a protective policy is stricter than the role's need.

The management account

The hold on `wf-management` is a role policy, because a service control policy never applies to the management account. The activation call removes it with the same command, and the account read at 10:12 EDT. We note it because it is the one account where a missing second step would look like success everywhere else.

The one-call activation

The approver runs one command, and it does everything the activation needs. The bank wrote and owns the script. It runs under the approver's role in the management account and reads the activation reference that counsel provides.

```bash
wf-ir-activate --reference DRILL-2026-08-12 --approver ciso
```

The command does five things and logs each to the bank's own audit trail: it checks that the reference matches counsel's signed statement of work on file; it detaches the hold policy `p-wrhold01` from the organization root `r-wrnf`; it detaches the same deny from the role in `wf-management`; it checks the removal directly, by reading the policies attached to the root (`organizations:ListPoliciesForTarget` on `r-wrnf`), walking `organizations:ListParents` from each account up to the root to read the policies on every parent organizational unit, and reading the role's inline policies in `wf-management`; and it writes the time to the activation log. A listing of one account's own attachments is not proof of its effective permissions, so the script does not stop there. The script does not call the role. The trust policy admits only Vantalect's principal, so the bank's approver cannot assume it, and the first real calls are Charles's: an authorized read that must succeed and an intentionally denied write that must be refused, which together show the effective result. A matching command, `wf-ir-hold`, puts the hold back; it was run at 11:05 EDT.

Who may run it: Mireille Danvers, and Imani Castell as the named alternate. Neither works for Vantalect, and neither needs Vantalect on the line. If neither can be reached, the hold stays on and the bank follows its own internal escalation procedure; the call sheet says so and names no third approver, because the bank has not given one authority to release the hold.

Compared with July 4

On July 4 the first read came 52 minutes after counsel signed the activation. In the drill it came 9 minutes after counsel's call. The two clocks do not start at exactly the same point, so we show the events rather than claim a single saving.

The incident role in the real incident and in the drill. Both clocks are Eastern Daylight Time.

Event	July 4	Drill, August 12
Counsel's activation signed	00:52 EDT	10:02 EDT
Charles on the bridge	00:58 EDT	Not applicable
Role available in the accounts	01:39 EDT to 01:40 EDT, by a self-managed StackSet the bank ran for the first time and a separate stack in `wf-management`	Already deployed; hold removed 10:05 EDT
First assume-role	01:44 EDT	10:07 EDT
First read	01:44 EDT (`EV-00614`)	10:09 EDT

In the incident, the role was deployed 42 minutes after Charles joined the bridge, between 00:58 EDT and 01:40 EDT. With the Retainer the role is already in place, and the time before the first read is the time the bank takes to sign a letter and run one command. Those 42 minutes are what the Retainer would have saved. We do not claim it would have changed what the attacker did in them: the containment policy was attached at 01:22 EDT, before the role existed, and Interceptor followed the sessions from 02:20 EDT.

What is open

- **Re-test.** We propose a re-run of this drill after any change to the StackSet, the hold, or a policy at the organization root, and once a quarter during the term. The bank decides.
- **The second responder.** The role admits one principal. If Vantalect names a backup responder, the trust policy changes and the drill repeats. Until then, the Retainer's commitments carry the one-responder limit.

- **Ownership of the hold.** The hold's source and the activation script live in the bank's policy repository. We recommend that Imani Castell schedule a test of the script itself, so it is exercised even when the drill is not.
- **Not tested.** A real incident with the bank's team under stress. That is the tabletop on Tuesday, September 15, 2026.

Notice

Wrenfield is fictional. Its people, customers, accounts, figures, and attacker are invented for this sample, and any resemblance to a real bank, company, person, or threat group is coincidental. The attacker's addresses come from ranges reserved for documentation.

© 2026 Vantalect, LLC. Sample deliverable; Wrenfield Bancorp, Inc. is fictional.

AWS, Amazon Web Services, and the AWS service names are trademarks of Amazon.com, Inc. or its affiliates. MITRE ATT&CK is a registered trademark of The MITRE Corporation. PostgreSQL is a registered trademark of the PostgreSQL Community Association of Canada. Other product names are trademarks of their owners. Vantalect is not affiliated with or endorsed by any of them. Wrenfield Bancorp, Inc. is fictional.