

Vantalect

AWS BREACH FORENSICS & RECOVERY · SAMPLE DELIVERABLE

Attack Narrative and Timeline

DISTRIBUTION: COUNSEL ONLY

Week 2 of 4: Trace · prepared for Wrenfield Bancorp, Inc.

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

July 24, 2026 · AWS Breach Forensics & Recovery: The Investigation and the Recovery (four weeks)

SAMPLE DELIVERABLE

Privileged and confidential. Prepared at the direction of counsel. Sample. Wrenfield is fictional.

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

About this report 2

Summary 3

Phases and techniques 5

Timeline 8

Before AWS: the stolen key 8

Discovery, Monday, June 29 to Thursday, July 2 8

Privilege use, July 3 evening 9

Impact, July 3 evening 11

Persistence and defense evasion 15

Exfiltration 16

Detection and response, July 3 22:44 to July 4 03:20 18

What the attacker tried and could not do 22

Attribution 25

Dwell and the intervals that matter 26

Open questions 27

What happens next 29

Notice 29

Appendix A: the complete timeline 30

About this report

This report sets out what the attacker did, in order, from the first use of a stolen key on Monday, June 29 to the last attempt on Saturday, July 4, and it gives the evidence for each step. It is Vantalect's finding as of Friday, July 24, 2026.

Distribution: Counsel only

Privileged and confidential. Prepared at the direction of counsel. The label does not protect a document by itself: privilege depends on how and why the work was commissioned, and counsel decides whether it applies. Vantalect makes no determination about privilege. We send this report to Diane Whitcombe of Tessaly & Brook LLP, outside counsel. Counsel decides who else reads it. The indicators in it are repeated in the companion report, Indicators of Compromise, which counsel may clear for release.

What the report rests on. The bank's organization trail recorded 519 calls that we attribute to the attacker, 224 of them denied. We read them from the copy preserved in the evidence account. Calls are grouped into rows when the same principal makes the same call in one burst (the row gives the count and the end time), so 164 rows stand for the 519 events. Each row carries evidence pointers (`EV-` numbers) into that copy.

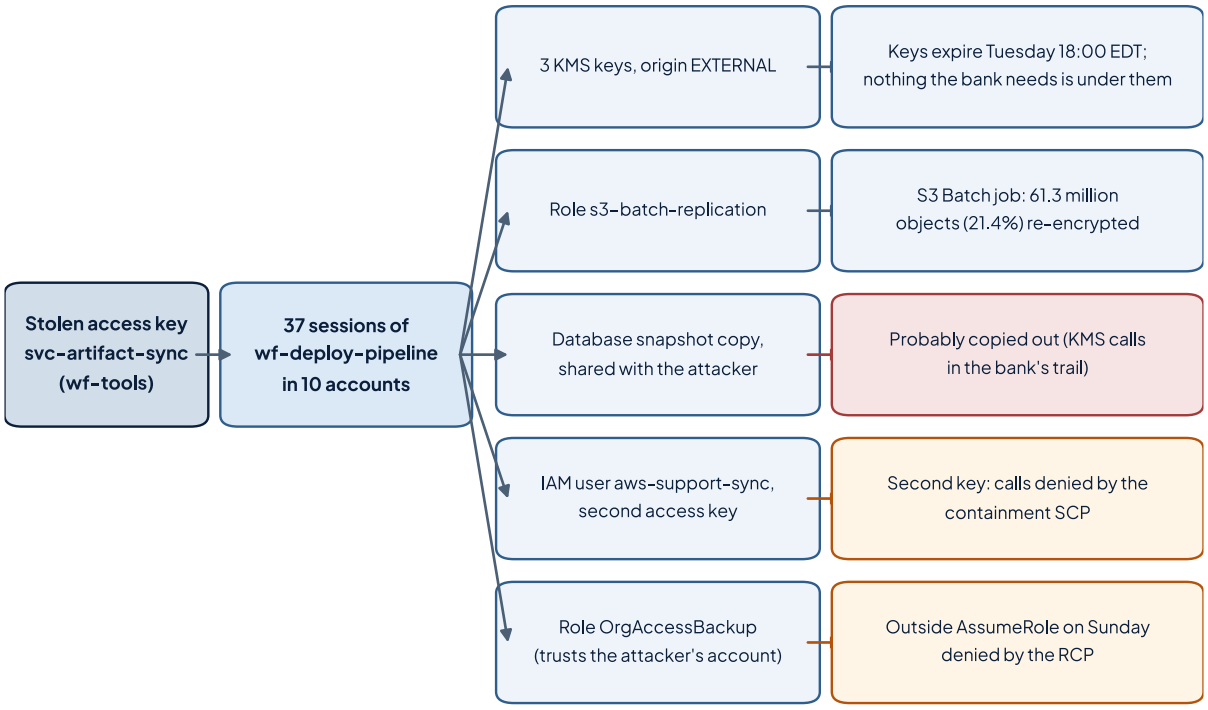
What it does not cover. It does not cover anything outside AWS. The theft of the key from a contractor's laptop comes from counsel's interview and is not Vantalect's evidence. It does not establish whether any regime applies, whether anything was a breach, or when a notice is due: those are decisions for the bank and counsel. It does not say who the attacker is beyond the wording in the attribution section. It does not cover object reads from the imaging archive, which were not logged (see the open questions), and it holds no finding about the bank's host systems: we found no host access.

All times are Eastern Daylight Time (EDT, UTC-4) unless a column says UTC. The July 3 holiday (Independence Day observed) and the weekend that followed matter to the story: the destructive operation began on a holiday evening.

Summary

An attacker used one stolen, never-rotated access key to enter the bank's AWS organization, spent 4 days, 17 hours, 50 minutes looking around, and then, on the evening of the July 3 holiday, tried to lock the bank out of its own data and demanded $14,500,000 in cryptocurrency by Tuesday, July 7 at 18:00 EDT. The key belonged to the IAM user `svc-artifact-sync` in the tools account and had not been rotated since March 14, 2023. From Monday, June 29, 03:14 EDT the attacker made read-only calls on four mornings, from two outside addresses, the first a residential proxy. At 21:04 EDT on Friday, July 3 the attacker assumed `wf-deploy-pipeline`, a role with administrator access in every workload account that trusted the whole tools account,

and made 35 more sessions in ten accounts (36 on July 3 and 4, or 37 including the July 1 test). Within three hours the attacker created 3 KMS keys with imported key material set to expire at the ransom deadline, started an S3 Batch job that re-encrypted 61,300,000 of the imaging archive's 286,000,000 objects (21.4%), deleted 9 database and 46 EBS snapshots, created a user and two roles to come back by, wrote ransom notes, and shared a copy of the loan database snapshot with the attacker's own account, `777788889999`. The account then made 7 KMS calls on the bank's key in 45 minutes, including a re-encryption to a key in the attacker's account, which is the pattern of the snapshot being copied out: **probable, with high confidence**, not confirmed. The bank, with Vantalect's captures taken first, removed the share at 02:08 EDT, 137 minutes after it was made, and by 02:10 EDT had dropped the outside statement from the three key policies and revoked the attacker's grant. The bank's containment service control policy went on at 01:22 EDT, 4 hours 18 minutes after the first write. After that, every attempt to create or destroy anything was denied, and the calls that succeeded were reads and new sessions, among them an administrator session in the attacker's `OrgAccessBackup` role at 01:36 that the policy did not stop. We found no host access, and the loan platform's logs show no attacker traffic. The evidence is consistent with a financially motivated actor that we track as `VTL-C1`; the attribution section gives the basis for that wording and its limits. No payment was made.



The chain from one stolen key to everything the attacker created, and where each branch ended.

The incident in numbers. Every figure is repeated, with its source, in the sections that follow.

Measure	Value	Source
First use of the stolen key	June 29, 03:14 EDT	EV-00003
First write-capable session	July 3, 21:04 EDT	EV-00083
Dwell, first use to first write	4 days, 17 hours, 50 minutes	Derived
First write to the containment SCP	4 hours 18 minutes	Derived
Attacker events, and how many were denied	519; 224	Organization trail
Sessions of `wf-deploy-pipeline` (the July 1 test and 36 from July 3), in how many accounts	37; 10	Organization trail
KMS keys created with imported material	3, expiring 2026-07-07 22:00:00 UTC	EV-00087
SSE-C attempts, all denied, in 40 seconds	214	EV-00098 to EV-00311
Imaging objects re-encrypted	61,300,000 of 286,000,000 (21.4%)	S3 Batch job
Database snapshots, then EBS snapshots, deleted	9; 46	Organization trail
KMS calls from the attacker's account on the lending key (the copy-out burst)	7; 7	EV-00522 to EV-00531; EV-00643
The snapshot share, made to its removal	137 minutes	EV-00520; EV-00632
Loan applicants' records in the shared snapshot	412,317	The bank's count

Phases and techniques

We map the attack to 8 phases of MITRE ATT&CK and to the AWS Customer Incident Response Team's (CIRT) threat technique catalog, and the mapping is exact for some phases and approximate for others; the last column says which. The identifiers T1486 (and the catalog's A001 to A003), T1485, T1490, and T1578 are cited as the AWS CIRT catalog and the ATT&CK matrix publish them. The other identifiers come from our general knowledge of the ATT&CK matrix. We have not checked any of them against the current published text for this report, and counsel to confirm them before this table goes outside counsel.

The attack by phase, with ATT&CK techniques. AWS CIRT identifiers (T1486.A001 to A003) are catalog names, not ATT&CK sub-techniques.

Phase	Window (EDT)	What the attacker did	Technique	Fit
Initial access	June 29, 03:14 EDT	GetCallerIdentity with the stolen long-lived key of svc-artifact-sync, from a residential proxy.	`T1078.004` Valid Accounts: Cloud Accounts	Direct. The theft is outside AWS and not our evidence, so it is not mapped.
Discovery	June 29, 03:14 EDT to July 2, 10:17 EDT	Read-only calls on four mornings (roles, keys, buckets, databases); ListAccounts denied each day. A second sweep of nine accounts on July 3.	`T1087.004` Account Discovery: Cloud Account; `T1580` Cloud Infrastructure Discovery	Approximate. The calls span several ATT&CK techniques; these two fit best.
Privilege use	July 1, 10:21 EDT to July 3, 21:04 EDT	AssumeRole of wf-deploy-pipeline (administrator access) in ten accounts: one test on July 1, then 36 sessions from 21:04 on July 3.	`T1078.004` Valid Accounts: Cloud Accounts; `T1548.005` Abuse Elevation Control Mechanism: Temporary Elevated Cloud Access	Approximate for T1548.005: the key's owner was already trusted for the role.
Impact	July 3, 21:06 EDT to July 4, 00:47 EDT	Three keys with expiring material, an SSE-C attempt, a lifecycle rule, snapshots copied under the attacker keys, an S3 Batch re-encryption, snapshots deleted, ransom notes.	`T1486` Data Encrypted for Impact (AWS CIRT: T1486.A001 S3 SSE-C, attempted; T1486.A002 EC2/EBS; T1486.A003 RDS); `T1485` Data Destruction; `T1490` Inhibit System Recovery; `T1578.001` Modify Cloud Compute Infrastructure: Create Snapshot; `T1657` Financial Theft	T1486 direct. The S3 Batch re-encryption fits no CIRT sub-technique we have read. T1657 is approximate.

Phase	Window (EDT)	What the attacker did	Technique	Fit
Persistence	July 3, 22:31 EDT to July 3, 22:36 EDT	User aws-support-sync with AdministratorAccess and a second key; role OrgAccessBackup trusting the attacker's account.	T1136.003 Create Account: Cloud Account; T1098.001 Account Manipulation: Additional Cloud Credentials; T1098.003 Account Manipulation: Additional Cloud Roles	Direct.
Defense evasion	July 3, 22:14 EDT to July 3, 22:15 EDT	DeleteTrail of a legacy trail (succeeded); StopLogging on the organization trail (denied). Names imitating the bank's own.	T1562.008 Impair Defenses: Disable or Modify Cloud Logs; T1036.005 Masquerading: Match Legitimate Name or Location	Direct for T1562.008. Approximate for T1036.005: we cannot say the names were chosen to deceive.
Exfiltration	July 3, 23:51 EDT to July 4, 00:51 EDT	A database snapshot copy shared with account 777788889999, then 7 KMS calls from that account on the bank's key (two grants, a key description, a re-encryption to a key in the attacker's account, two decryptions, a grant retired).	T1537 Transfer Data to Cloud Account	The technique fits. The copy-out is inferred from the KMS calls and the share: probable, not seen. The share was removed at 02:08.
Response (the attacker's reaction to containment)	July 4, 01:23 EDT to July 4, 02:59 EDT	After the SCP at 01:22: new sessions, an administrator session in OrgAccessBackup at 01:36 that the SCP did not block, then denied attempts to create a key, delete key material, and delete the database. The attacker's account may have tried the lending key again after July 4. A request that the bank's key policy or the resource control policy denies is logged only in the caller's account, so the bank's trail cannot show it either way.	T1078.004 Valid Accounts: Cloud Accounts; T1485 Data Destruction (attempted); T1578.003 Modify Cloud Compute Infrastructure: Delete Cloud Instance (attempted)	Approximate. These are attempts, and each was denied or had no effect.

The phases are not strictly sequential. The attacker returned to discovery for the other nine workload accounts on the night of July 3, between the first keys and the snapshots, and the persistence steps came in the middle of the impact steps. The timeline below follows the clock; this table follows the technique.

Timeline

The attack ran in four blocks: four mornings of quiet reading, 4 hours 18 minutes of destruction on the holiday evening, a burst of attempts after containment, and one last attempt two days later. This section shows the rows that matter for each phase. Appendix A, on landscape pages, has every one of the 164 rows in time order. A row ID such as `AT-053` is the same in both.

Before AWS: the stolen key

The first step is not in any AWS log and is not Vantalect's evidence. An infostealer on a contractor's personal laptop takes ~/.aws/credentials, which holds svc-artifact-sync's key (from counsel's interview; outside AWS). The date is Monday, June 22, 2026, as counsel gave it to us. We rely on it only for the order of events: the key was stolen before its first use on Monday, June 29.

Discovery, Monday, June 29 to Thursday, July 2

For four days the attacker only read. Each morning between 02:00 and 11:00 EDT (06:00 to 15:00 UTC) the stolen key listed roles, keys, buckets, and databases in the tools account. The first two days came from `198.51.100.23`; the next two from `198.51.100.41`. These were read-only management events. They reached the organization trail and raised no alert, because read-only events were not routed to the bank's security operations. On Wednesday, July 1 the attacker found the trust policy of `wf-deploy-pipeline` and assumed it once in `wf-nonprod-data` as a test, then listed buckets, databases, and keys there (EV-00051 to EV-00054). After a last morning of reading on Thursday, July 2, 2026, the attacker went quiet until the evening of July 3.

Selected discovery rows: 8 of the 52 rows before July 3. The four ListAccounts calls were all denied. Times are MM-DD HH:MM:SS; account names drop the prefix `wf-`.

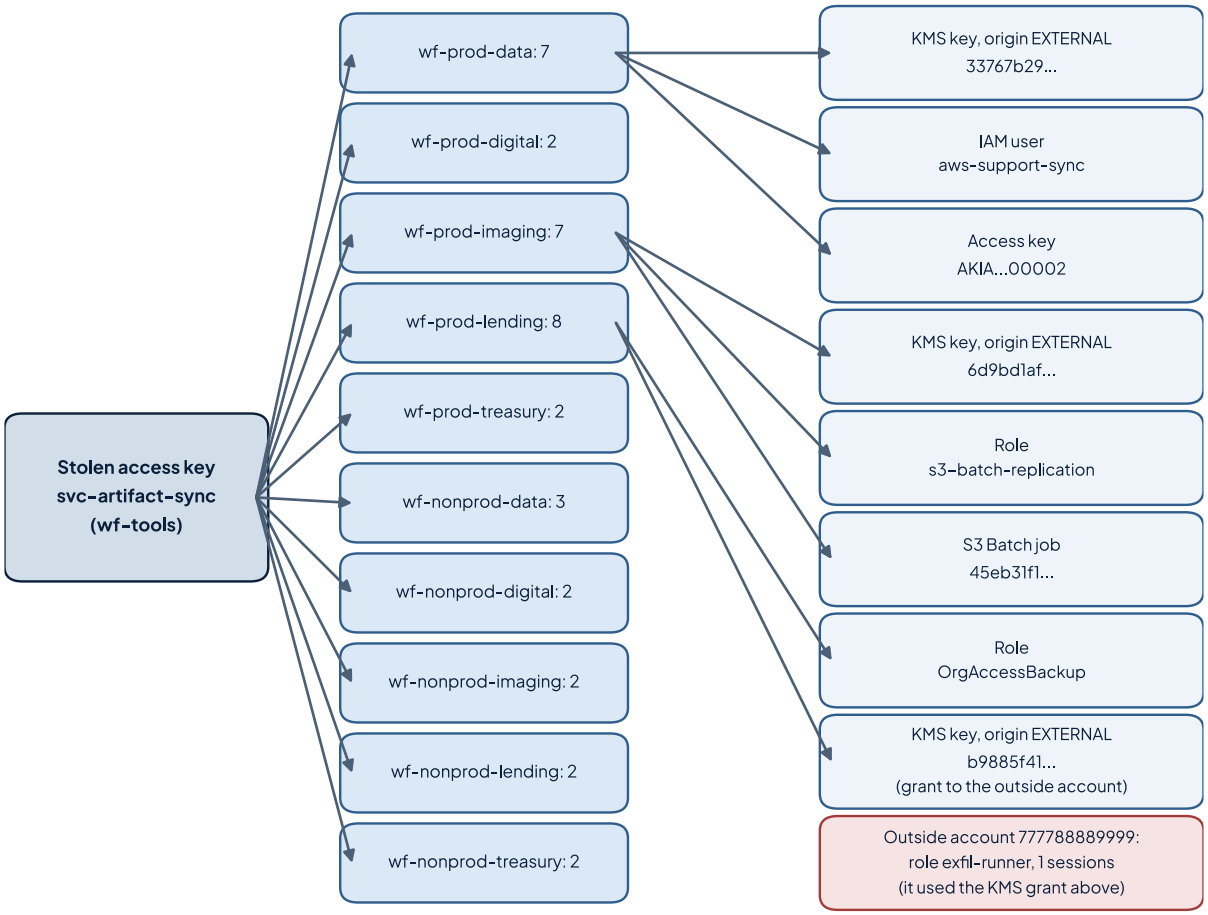
Row	Local (EDT)	UTC	Account	Principal and session	API call and result	Evidence
AT-001	06-29 03:14:02	06-29 07:14:02	`tools`	`svc-artifact-sync`	`GetCallerIdentity`	`EV-00003`
AT-002	06-29 03:20:27	06-29 07:20:27	`tools`	`svc-artifact-sync`	`ListAccounts`; Denied: AccessDeniedException	`EV-00004`
AT-006	06-29 06:20:16	06-29 10:20:16	`tools`	`svc-artifact-sync`	`DescribeDBInstances`	`EV-00008`
AT-010	06-29 08:00:24	06-29 12:00:24	`tools`	`svc-artifact-sync`	`ListRoles`	`EV-00012`
AT-025	07-01 02:00:00	07-01 06:00:00	`tools`	`svc-artifact-sync`	`GetCallerIdentity`	`EV-00033`
AT-037	07-01 10:21:14	07-01 14:21:14	`nonprod-data`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00051`
AT-038	07-01 10:22:05	07-01 14:22:05	`nonprod-data`	`jenkins-sync-4440`	`ListBuckets`	`EV-00052`
AT-052	07-02 10:17:34	07-02 14:17:34	`tools`	`svc-artifact-sync`	`ListAccessKeys` on `svc-artifact-sync`	`EV-00068`

Privilege use, July 3 evening

The administrator role did the rest of the work. At 21:04 EDT on Friday, July 3 the same key assumed `wf-deploy-pipeline` in `wf-prod-lending` as session `jenkins-sync-4471`, from `203.0.113.47`. Within the hour the attacker had sessions in all five production accounts and, in a sweep from 21:20 EDT to 21:31 EDT, in the digital and treasury accounts and in all five non-production accounts, where it made only discovery calls and no writes. The sessions last an hour, so the attacker kept assuming the role again, about every hour, until 01:24 EDT. Each AssumeRole is recorded twice, in the role's account and in the tools account. The session names `jenkins-sync-NNNN` imitate the bank's own build sessions (`jenkins-build-` and `artifact-sync-`).

Selected privilege-use rows. Rows AT-053 to AT-154 hold the other AssumeRole calls; the appendix lists them all. Times are MM-DD HH:MM:SS; account names drop the prefix `wf-`.

Row	Local (EDT)	UTC	Account	Principal and session	API call and result	Evidence
AT-053	07-03 21:04:11	07-04 01:04:11	`prod-lending`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00083`
AT-058	07-03 21:09:02	07-04 01:09:02	`prod-imaging`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00088`
AT-061	07-03 21:09:40	07-04 01:09:40	`prod-data`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00091`
AT-070	07-03 21:20:05	07-04 01:20:05	`prod-digital`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00313`
AT-079	07-03 21:26:00	07-04 01:26:00	`nonprod-lending`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00322`
AT-093	07-03 21:30:00	07-04 01:30:00	`nonprod-treasury`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00336`
AT-110	07-03 21:56:30	07-04 01:56:30	`prod-lending`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00441`



Account: sessions of wf-deploy-pipeline (37 in 10 accounts)

The session tree, drawn from Interceptor's replay of the logs: each box is an account, with the number of sessions of `wf-deploy-pipeline` the attacker opened there, and each item on the right sits under the account whose session created it.

Impact, July 3 evening

In 2 hours 24 minutes, from the first key to the ransom notes, the attacker tried four ways to take the data away from the bank. One failed and three worked. They are set out in the order the attacker tried them.

First, the keys and the playbook that failed

At 21:06 EDT the attacker created a KMS key in `wf-prod-lending` with origin `EXTERNAL`, gave it the alias `alias/wf-backup-2026`, and imported key material valid until 2026-07-07 22:00:00 UTC, which is Tuesday, July 7 at 18:00 EDT, the ransom deadline. Three minutes later the attacker repeated this in `wf-prod-imaging` and `wf-prod-data`. Each key policy kept the account root and

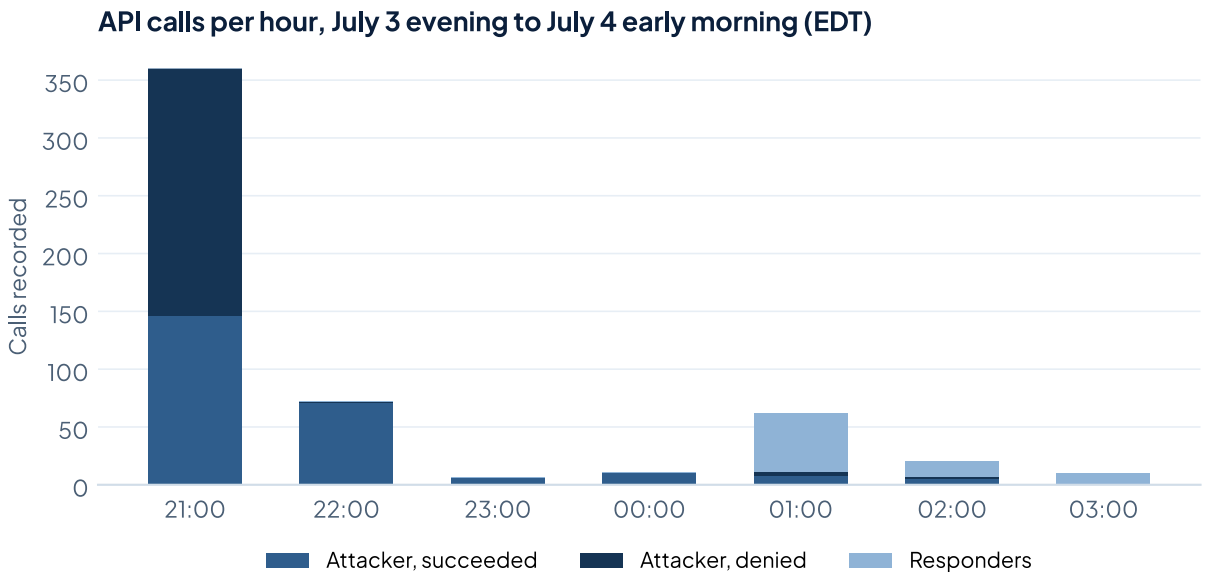
added a statement for account 777788889999. Keys that expire on their own need no further call to lock the data. At 21:12 EDT the attacker tried the older playbook, copying data-lake objects in place with a customer-supplied key (SSE-C): 214 requests in 40 seconds, every one denied, because the bucket blocks SSE-C. At 21:14 EDT the attacker set a lifecycle rule on the imaging archive to expire noncurrent versions after one day.

The keys, the SSE-C attempt, and the lifecycle rule. Times are MM-DD HH:MM:SS; account names drop the prefix `wf-`.

Row	Local (EDT)	UTC	Account	Principal and session	API call and result	Evidence
AT-054	07-03 21:06:08	07-04 01:06:08	`prod-lending`	`jenkins-sync-4471`	`CreateKey` on `key/b9885f41-4ff8-5cbf-9…`	`EV-00084`
AT-057	07-03 21:07:00	07-04 01:07:00	`prod-lending`	`jenkins-sync-4471`	`ImportKeyMaterial` on `b9885f41-4ff8-5cbf-9f54-…`	`EV-00087`
AT-059	07-03 21:09:21	07-04 01:09:21	`prod-imaging`	`jenkins-sync-4473`	`CreateKey` on `key/6d9bd1af-48a0-5150-b…`	`EV-00089`
AT-063	07-03 21:09:55	07-04 01:09:55	`prod-data`	`jenkins-sync-4475`	`CreateKey` on `key/33767b29-2aaa-5253-8…`	`EV-00093`
AT-067	07-03 21:10:47	07-04 01:10:47	`prod-data`	`jenkins-sync-4475`	`ImportKeyMaterial` on `33767b29-2aaa-5253-83bf-…`	`EV-00097`
AT-068	07-03 21:12:03	07-04 01:12:03	`prod-data`	`jenkins-sync-4475`	`CopyObject` (214 calls, to 21:12) on `wf-datalake-prod`; Denied: AccessDenied	`EV-00098` to `EV-00311`
AT-069	07-03 21:14:26	07-04 01:14:26	`prod-imaging`	`jenkins-sync-4473`	`PutBucketLifecycleConfiguration` on `wf-imaging-archive-prod`	`EV-00312`

Second, the copies and the batch job

The attacker took a snapshot of the loan database `wf-los-prod` at 21:22 EDT, snapshotted 46 EBS volumes at 21:35 EDT, and copied each under an attacker key. At 21:39 EDT it created the role `s3-batch-replication`, and at 21:41 EDT an S3 Batch Operations job that copies every object in the imaging archive in place, encrypted with the imaging attacker key. Each copy becomes the current version, and the bank's version becomes noncurrent. Versioning was on, and that is why the bank could recover. At 21:58 EDT the attacker copied the database snapshot under the prod-lending attacker key. The job ran for 227 minutes (3 hours 47 minutes) and the bank canceled it at 01:28 EDT, with 61,300,000 objects (21.4%) re-encrypted.



Recorded API calls per hour: the attacker's allowed and denied calls, and the responders'. Counts are the calls inside the collapsed rows.

Snapshots, the copy job, and the role it ran as. Times are MM-DD HH:MM:SS; account names drop the prefix `wf-`.

Row	Local (EDT)	UTC	Account	Principal and session	API call and result	Evidence
AT-074	07-03 21:22:17	07-04 01:22:17	`prod-lending`	`jenkins-sync-4471`	`CreateDBSnapshot` on `wf-los-prod`	`EV-00317`
AT-101	07-03 21:35:10	07-04 01:35:10	`prod-lending`	`jenkins-sync-4494`	`CreateSnapshot` (31 calls, to 21:36) on `snapshot/snap-0d8ae7cbff…`	`EV-00344` to `EV-00389`
AT-102	07-03 21:35:22	07-04 01:35:22	`prod-data`	`jenkins-sync-4496`	`CreateSnapshot` (15 calls, to 21:35) on `snapshot/snap-09f30351bb…`	`EV-00351` to `EV-00379`
AT-104	07-03 21:39:12	07-04 01:39:12	`prod-imaging`	`jenkins-sync-4499`	`CreateRole` on `s3-batch-replication`	`EV-00391`
AT-106	07-03 21:41:00	07-04 01:41:00	`prod-imaging`	`jenkins-sync-4499`	`CreateJob` on `role/s3-batch-replication`	`EV-00393`
AT-108	07-03 21:47:00	07-04 01:47:00	`prod-lending`	`jenkins-sync-4494`	`CopySnapshot` (31 calls, to 21:48) on `snapshot/snap-0fbdf84542…`	`EV-00395` to `EV-00440`
AT-109	07-03 21:47:01	07-04 01:47:01	`prod-data`	`jenkins-sync-4496`	`CopySnapshot` (15 calls, to 21:47) on `snapshot/snap-091f899d61…`	`EV-00396` to `EV-00424`
AT-111	07-03 21:58:31	07-04 01:58:31	`prod-lending`	`jenkins-sync-4502`	`CopyDBSnapshot` on `wf-los-prod-bk-20260703`	`EV-00442`

Third, the deletions, the notes, and the database

At 22:05 EDT the attacker deleted the 9 month-end database snapshots in `wf-prod-lending` and, at 22:09 EDT, the 46 bank EBS snapshots in the lending and data accounts. The EBS snapshots went into the Recycle Bin, which the attacker did not check. At 23:30 EDT it wrote `RESTORE-INSTRUCTIONS.txt` into the imaging and data-lake buckets; a third note in the lending documents bucket is in no log, because that bucket has no data events, and we know it from the object listing (23:30 EDT). At 00:47 EDT, after the bank's incident had begun, the attacker turned off deletion protection on `wf-los-prod`.

The deletions, two of the three ransom notes, and the deletion-protection change. Times are MM-DD HH:MM:SS; account names drop the prefix `wf-`.

Row	Local (EDT)	UTC	Account	Principal and session	API call and result	Evidence
AT-113	07-03 22:05:03	07-04 02:05:03	`prod-lending`	`jenkins-sync-4505`	`DeleteDBSnapshot` (9 calls, to 22:05) on `wf-los-prod-monthend-202…`	`EV-00444` to `EV-00452`
AT-115	07-03 22:09:04	07-04 02:09:04	`prod-lending`	`jenkins-sync-4505`	`DeleteSnapshot` (31 calls, to 22:09) on `snap-02b6db40772885b3f`	`EV-00454` to `EV-00485`
AT-117	07-03 22:09:40	07-04 02:09:40	`prod-data`	`jenkins-sync-4510`	`DeleteSnapshot` (15 calls, to 22:09) on `snap-0e136e7b45205b0d1`	`EV-00486` to `EV-00500`
AT-135	07-03 23:30:11	07-04 03:30:11	`prod-imaging`	`jenkins-sync-4530`	`PutObject` on `wf-imaging-archive-prod`	`EV-00518`
AT-136	07-03 23:30:36	07-04 03:30:36	`prod-data`	`jenkins-sync-4532`	`PutObject` on `wf-datalake-prod`	`EV-00519`
AT-145	07-04 00:47:09	07-04 04:47:09	`prod-lending`	`jenkins-sync-4534`	`ModifyDBInstance` on `wf-los-prod`	`EV-00529`

Persistence and defense evasion

The attacker made a user, a second key, and a role to come back by, and tried to turn off the logs. At 22:31 EDT it created the IAM user `aws-support-sync` in `wf-prod-data`, attached `AdministratorAccess`, and made the second key `AKIAWRFNEXAMPLE00002`. At 22:36 EDT it created the role `OrgAccessBackup` in `wf-prod-lending`, trusting account `777788889999`. The user and role names imitate a vendor support account and a backup role. At 22:14 EDT it deleted a legacy account trail, which succeeded, and a minute later tried to stop the organization trail, which a member account cannot do. The organization trail kept recording. The legacy trail itself is gone; the organization trail records the same activity.

Persistence (AT-127 to AT-131) and defense evasion (AT-118, AT-119). Times are MM-DD HH:MM:SS; account names drop the prefix `wf-`.

Row	Local (EDT)	UTC	Account	Principal and session	API call and result	Evidence
AT-127	07-03 22:31:06	07-04 02:31:06	`prod-data`	`jenkins-sync-4510`	`CreateUser` on `aws-support-sync`	`EV-00510`
AT-128	07-03 22:31:31	07-04 02:31:31	`prod-data`	`jenkins-sync-4510`	`AttachUserPolicy` on `aws-support-sync`	`EV-00511`
AT-129	07-03 22:31:58	07-04 02:31:58	`prod-data`	`jenkins-sync-4510`	`CreateAccessKey` on `aws-support-sync`	`EV-00512`
AT-130	07-03 22:36:14	07-04 02:36:14	`prod-lending`	`jenkins-sync-4505`	`CreateRole` on `OrgAccessBackup`	`EV-00513`
AT-131	07-03 22:36:40	07-04 02:36:40	`prod-lending`	`jenkins-sync-4505`	`AttachRolePolicy` on `OrgAccessBackup`	`EV-00514`
AT-118	07-03 22:14:18	07-04 02:14:18	`prod-lending`	`jenkins-sync-4505`	`DeleteTrail` on `wf-lending-legacy-trail`	`EV-00501`
AT-119	07-03 22:15:04	07-04 02:15:04	`prod-lending`	`jenkins-sync-4505`	`StopLogging` on `trail/wf-org-trail`; Denied: AccessDeniedException	`EV-00502`

Exfiltration

The attacker shared a copy of the loan database snapshot with its own account, and its account then used the key that protects the copy. At 23:51 EDT the attacker called `ModifyDBSnapshotAttribute`, adding account `777788889999` to the copy `wf-los-prod-bk-20260703`. Fifteen minutes later, from 00:06:09 EDT to 00:51:06 EDT, the role `exfil-runner` in that account made 7 KMS calls on the prod-lending attacker key. All were logged in the bank's trail because the key is the bank's. In order:

- 00:06:09 EDT: `CreateGrant` naming `exfil-runner` as grantee, and at 00:06:37 EDT a second `CreateGrant`; between them, at 00:06:21 EDT, a `DescribeKey` (`EV-00522` to `EV-00523`).
- The second grant names the database service as grantee and the attacker's role as **retiring principal**, so the attacker's role could end a grant the database service held (`EV-00524`).
- 00:07:02 EDT: `ReEncrypt` from the bank's key to a **key in account `777788889999`**, through the database service (`EV-00525`).
- 00:28:14 EDT and 00:50:48 EDT: two `Decrypt` calls with the database snapshot's identifier in the encryption context.
- 00:51:06 EDT: `RetireGrant`, which ended the grant.

Why this is probable, and why it is not confirmed. The three facts that carry it are the re-encryption toward a key in the attacker's account, the grant that lets the attacker's role retire a grant held by the database service, and the share, which was open when the calls were made. Together they are the shape we would expect on the source key during a cross-account snapshot copy; we have not tested that against a live copy. The copy itself happens in account `777788889999`, outside our view, so **we put the copy-out at probable, with high confidence, and we do not call it confirmed.** The logs cannot show whether a copy finished, how much the re-encryption moved, what the destination key now protects, or whether anyone read the data.

The share stayed open 137 minutes, from 23:51:12 EDT to 02:08:20 EDT. The bank closed the path in stage 1 of the response, on July 4: Vantalect captured the snapshot's attributes, the three key policies, and the grants first (`EV-00625` to `EV-00631`), then the bank's break-glass role removed the account from the snapshot and replaced the three key policies without the outside statement, and revoked the attacker's grant at 02:10:30 EDT. One more call came from the same account on Saturday, July 4 at 02:59:30 EDT: a `Decrypt` on the same key from `exfil-runner` at `203.0.113.47`, denied by the resource control policy (`explicit deny in a resource control policy`). With it the account made 7 KMS calls on bank keys, from July 4, 00:06 EDT to July 4, 00:51 EDT. KMS records an operation on a key in another account in both accounts' trails. We assume, and have not verified, that a denied call reaches the key owner's trail the same way; that is why the row appears in the bank's trail. The snapshot held 412,317 loan applicants' records: New York 389,904, Pennsylvania 15,077, other states 7,336.

The share and the copy-out burst. Times are MM-DD HH:MM:SS; account names drop the prefix `wf-`.

Row	Local (EDT)	UTC	Account	Principal and session	API call and result	Evidence
AT-137	07-03 23:51:12	07-04 03:51:12	`prod-lending`	`jenkins-sync-4527`	`ModifyDBSnapshotAttribute` on `wf-los-prod-bk-20260703`	`EV-00520`
AT-139	07-04 00:06:09	07-04 04:06:09	`prod-lending`	`s-9172` (exfil-runner, account 777788889999)	`CreateGrant` (2 calls, to 00:06) on `key/b9885f41-4ff8-5cbf-9…`	`EV-00522` to `EV-00524`
AT-140	07-04 00:06:21	07-04 04:06:21	`prod-lending`	`s-9172` (exfil-runner, account 777788889999)	`DescribeKey` on `key/b9885f41-4ff8-5cbf-9…`	`EV-00523`
AT-141	07-04 00:07:02	07-04 04:07:02	`prod-lending`	`s-9172` (exfil-runner, account 777788889999)	`ReEncrypt` on `key/b9885f41-4ff8-5cbf-9…`	`EV-00525`
AT-144	07-04 00:28:14	07-04 04:28:14	`prod-lending`	`s-9172` (exfil-runner, account 777788889999)	`Decrypt` on `key/b9885f41-4ff8-5cbf-9…`	`EV-00528`
AT-146	07-04 00:50:48	07-04 04:50:48	`prod-lending`	`s-9172` (exfil-runner, account 777788889999)	`Decrypt` on `key/b9885f41-4ff8-5cbf-9…`	`EV-00530`
AT-147	07-04 00:51:06	07-04 04:51:06	`prod-lending`	`s-9172` (exfil-runner, account 777788889999)	`RetireGrant` on `key/b9885f41-4ff8-5cbf-9…`	`EV-00531`

Detection and response, July 3 22:44 to July 4 03:20

The incident record holds one GuardDuty finding, raised 1 hour 40 minutes after the first write, and a person was paged 34 minutes after that. GuardDuty raised `Persistence:IAMUser/AnomalousBehavior` (severity 8.0) in `wf-audit` at 22:44 EDT, 1 hour 40 minutes after the first write; Security Hub forwarded it to the managed security service provider, which paged the on-call engineer, Owen Tribble, at 23:18 EDT, 34 minutes later, on a holiday. The bank signed the emergency activation at 00:52 EDT, and the containment SCP went on at 01:22 EDT.

How the provider triaged it. The record shows the finding, the forward, and the page, and nothing between them: we hold no ticket, runbook entry, or note from the provider for those 34 minutes. What the record does show is the cost. By the time of the finding the S3 Batch job had run for more than an hour, the database snapshots and EBS snapshots had been deleted, and the new user and role existed. The page came before the ransom notes were written at 23:30 EDT and

before the snapshot was shared at 23:51 EDT. One engineer was paged; there was no second person to page and no rule that sent a high-severity persistence finding to a phone at once. The open questions ask the provider for its queue.

Assumption, not a record: what GuardDuty would be expected to raise

GuardDuty documents finding types that the earlier activity matches. We have no record that any of them was raised, and none is in the incident record. If they were raised, they reached the provider's queue before 22:44, and the detection gap in the root-cause report (RC-5) is sharper: signals existed and were not acted on. If they were not, the gap is the same one RC-5 already names.

Finding types GuardDuty would be expected to raise for the activity. An assumption, labeled as one; not events in the record.

Activity	Finding type	Severity as documented, and our caveat
DeleteTrail of the legacy trail at 22:14 and the denied StopLogging at 22:15 (AT-118, AT-119)	`Stealth:IAMUser/CloudTrailLoggingDisabled`	Documented as Low severity. It would have been raised within minutes of the call, 30 minutes before the finding the provider paged on.
Four days of read-only discovery on a three-year-old service key from addresses that were not the bank's NAT (AT-001 to AT-052)	`Discovery:IAMUser/AnomalousBehavior`	Documented as Low severity. A service key that has run only from the bank's NAT is the baseline such a finding is built on.
The whole sequence: discovery, a role assumed in ten accounts, keys, snapshots, deletions, and a new user and role	`AttackSequence:IAM/CompromisedCredentials`	Documented as Critical severity; GuardDuty Extended Threat Detection launched on December 1, 2024. We have not confirmed when the IAM attack-sequence type appeared or whether it was enabled for the bank.

Detections, calls, and decisions around the attack, from the incident timeline (MM-DD HH:MM:SS). The first row is outside AWS and is not Vantalect's evidence.

Local (EDT)	UTC	What happened	Who
06-22 12:00:00	06-22 16:00:00	An infostealer on a contractor's personal laptop takes ~/.aws/ credentials, which holds svc-artifact-sync's key (from counsel's interview; outside AWS) (not in the trail)	Attacker
07-03 22:44:20	07-04 02:44:20	GuardDuty raises Persistence:IAMUser/AnomalousBehavior in wf-audit; Security Hub forwards it to the managed security service provider	AWS
07-03 23:18:00	07-04 03:18:00	The provider pages Owen Tribble (34 minutes after the finding; a holiday on-call)	Security provider
07-03 23:30:50	07-04 03:30:50	RESTORE-INSTRUCTIONS.txt written to the lending documents bucket (not in the trail: that bucket has no data events) (not in the trail)	Attacker
07-04 00:05:00	07-04 04:05:00	Owen confirms the note and calls Imani Castell and Mireille Danvers	Bank
07-04 00:52:00	07-04 04:52:00	Emergency activation signed: counsel's three-party letter and the incident statement of work	Counsel
07-04 00:58:00	07-04 04:58:00	Charles on the bridge	Vantalect
07-04 01:50:00	07-04 05:50:00	Loan origination taken offline by the bank to freeze its state	Bank
07-04 02:20:00	07-04 06:20:00	Interceptor live in wf-audit, seeded with the first key: it follows the 37 wf-deploy-pipeline sessions and finds the user, key, and two roles created	Vantalect
07-04 02:50:00	07-04 06:50:00	Interceptor alert: AKIAWRFNEXAMPLE00002 active from 203.0.113.47	Vantalect
07-05 11:15:00	07-05 15:15:00	NYDFS notified	Bank
07-07 18:00:00	07-07 22:00:00	The imported key material expires on the three attacker keys; nothing the bank needs is under them	AWS

The next table shows the defenders' own calls in the trail. They are the calls that changed what the attacker could do.

Selected defender calls that changed what the attacker could do (MM-DD HH:MM:SS EDT; accounts without the prefix `wf-`). The sessions were revoked in all ten workload accounts (`EV-00555` to `EV-00564`).

Row	Local (EDT)	Account	Principal	API call	Evidence
RE-003	07-04 01:22:15	management	imani.castell@wrenfield.example	AttachPolicy	EV-00537
RE-004	07-04 01:24:30	tools	imani-castell	UpdateAccessKey on svc-artifact-sync	EV-00543
RE-005	07-04 01:26:05	prod-lending	imani-castell	PutRolePolicy on wf-deploy-pipeline	EV-00555
RE-015	07-04 01:28:05	prod-imaging	imani-castell	UpdateJobStatus on job/45eb31f1-d889-53b5-a…	EV-00565
RE-016	07-04 01:31:40	management	imani.castell@wrenfield.example	CreateStackSet	EV-00567
RE-017	07-04 01:35:12	prod-imaging	imani-castell	DeleteBucketLifecycle on wf-imaging-archive-prod	EV-00568
RE-052	07-04 02:03:10	prod-lending	charles-sieg	DescribeDBSnapshotAttributes on wf-los-prod-bk-20260703	EV-00625
RE-059	07-04 02:08:20	prod-lending	imani-castell	ModifyDBSnapshotAttribute on wf-los-prod-bk-20260703	EV-00632
RE-060	07-04 02:09:05	prod-lending	imani-castell	PutKeyPolicy on key/b9885f41-4ff8-5cbf-9…	EV-00633
RE-061	07-04 02:09:25	prod-imaging	imani-castell	PutKeyPolicy on key/6d9bd1af-48a0-5150-b…	EV-00634
RE-062	07-04 02:09:45	prod-data	imani-castell	PutKeyPolicy on key/33767b29-2aaa-5253-8…	EV-00635
RE-063	07-04 02:10:30	prod-lending	imani-castell	RevokeGrant on key/b9885f41-4ff8-5cbf-9…	EV-00636
RE-064	07-04 02:58:12	management	imani.castell@wrenfield.example	UpdatePolicy	EV-00642
RE-065	07-04 03:01:10	prod-data	imani-castell	UpdateAccessKey on aws-support-sync	EV-00646
RE-070	07-04 03:16:40	audit	imani-castell	ListFindingsV2 on analyzer/wf-org-external…	EV-00652

Row	Local (EDT)	Account	Principal	API call	Evidence
RE-072	07-04 03:20:14	`management`	`imani.castell@wrenfield.example`	`AttachPolicy`	`EV-00654`

Three groups of these calls need a sentence each. **The closure of the exfiltration path (stage 1).** Vantalect's `VantalectIncidentResponse` role captured the snapshot's attributes at 02:03:10 EDT, then the three key policies and the three sets of grants, until 02:05:00 EDT. Then the bank's `wf-break-glass` role removed the account from the snapshot (02:08:20 EDT), replaced the three key policies at 02:09:05 EDT, 02:09:25 EDT, and 02:09:45 EDT, and revoked the attacker's grant (02:10:30 EDT). The database service's own grant had already been retired at 00:51:06 EDT. **The deploy of the incident role.** The bank used a self-managed StackSet whose execution role is `wf-break-glass`, which the SCP exempts; the management account got its copy from a separate `CreateStack` (01:40:25 EDT), and the incident role reached the accounts 42 minutes after Charles joined the bridge. **The resource control policy.** The bank enabled the policy type at 03:15:10 EDT, listed the outside trusts with Access Analyzer at 03:16:40 EDT, and attached `wf-deny-outside-access` at 03:20:14 EDT. The listing found two outside trusts: the attacker's `OrgAccessBackup` trust, which the policy cuts, and Vantalect's incident role, which it exempts; the bank had no other.

The containment SCP is the control that mattered most. It denied key import and deletion, snapshot deletion and sharing, database deletion and modification, lifecycle and trail changes, IAM user, key, and role creation, and S3 Batch job creation, for every principal except the incident role and the bank's break-glass role. The bank attached it 4 hours 18 minutes after the first write. The next section shows what it stopped.

What the attacker tried and could not do

10 rows of the attacker's activity were denied, by five different controls, and the table says what stopped each. The 214 SSE-C denials are one row. Three controls were in place before the attack: the user's missing Organizations permission, the SSE-C block on the bucket, and the organization trail's protection from member accounts. Two were added by the bank on July 4: the containment SCP and the resource control policy. The first denials did not stop the attack. They show what the attacker was after: the account list, a way to overwrite data without a bank-held key, and the logs.

Every denied row in the attacker's activity (MM-DD HH:MM:SS EDT). Rows AT-002, AT-017, AT-026, and AT-048 are the same ListAccounts call on four mornings.

Row	Local (EDT)	Call	What the attacker tried	What stopped it	Evidence
AT-002	06-29 03:20:27	`ListAccounts` in `tools`	List the organization's accounts from the stolen user	The user has no Organizations permission, so the attacker never saw the account list and had to find the deploy role by trial.	`EV-00004`
AT-017	06-30 04:08:57	`ListAccounts` in `tools`	The same call, June 30	The same missing permission.	`EV-00021`
AT-026	07-01 02:08:00	`ListAccounts` in `tools`	The same call, July 1	The same missing permission.	`EV-00036`
AT-048	07-02 06:32:38	`ListAccounts` in `tools`	The same call, July 2	The same missing permission.	`EV-00064`
AT-068	07-03 21:12:03	`CopyObject` in `prod-data` (214 denied)	Overwrite data-lake objects with a key only the attacker holds (SSE-C), the 2025 playbook	The bucket blocks SSE-C (AWS disabled it for accounts without SSE-C objects in April 2026), so every request returned 403. No data-lake object was changed.	`EV-00098` to `EV-00311`
AT-119	07-03 22:15:04	`StopLogging` in `prod-lending`	Stop logging on the organization trail from a member account	A member account cannot change an organization trail. The trail recorded everything that followed, including this attempt.	`EV-00502`
AT-155	07-04 01:24:35	`CreateAccessKey` in `prod-data`	Create another access key for aws-support-sync, seconds after the stolen key was deactivated (EV-00543)	The containment SCP, attached two minutes earlier, denied IAM user and key creation.	`EV-00544`

Row	Local (EDT)	Call	What the attacker tried	What stopped it	Evidence
AT-156	07-04 01:31:14	`DeleteImportedKeyMaterial` in `prod-data`	Delete the imported key material on the data-lake key, from the second key's user in the key's own account (KMS allows the call only there). It is a lock-now move: the material is gone at once, and only the attacker holds the copy that could restore it	The containment SCP (explicit deny in a service control policy), nine minutes after it was attached.	`EV-00566`
AT-158	07-04 01:39:05	`DeleteDBInstance` in `prod-lending`	Delete the production loan database with no final snapshot, from the OrgAccessBackup administrator session that had been opened at 01:36	The containment SCP, which denied this call and not the session. Deletion protection had been turned off at 00:47 (EV-00529), so the SCP was the only barrier we can see in the trail.	`EV-00570`
AT-164	07-04 02:59:30	`ListUsers` in `prod-data` (2 denied)	List IAM users with the second key	The tightened SCP of 02:58, which denies everything to the attacker's principals.	`EV-00643` to `EV-00644`

Three further points matter. The attacker's deletion of the legacy trail succeeded, and the register carries it as accepted: the organization trail records the same API activity. And the post-containment AssumeRole calls at 01:23 and 01:24 succeeded (`EV-00538` to `EV-00542`): the SCP does not deny AssumeRole, so the deploy role itself stayed usable until the bank deactivated the key at 01:24 and revoked the sessions at 01:26. The containment worked by denying what the sessions could create and destroy, not by blocking the sessions. The third point is the same lesson at a larger scale. At 01:36 EDT, the `exfil-runner` session from the attacker's account assumed `OrgAccessBackup` in `wf-prod-lending` and the call succeeded (`EV-00569`): the role carries `AdministratorAccess` (attached at 22:36 EDT), the SCP does not deny AssumeRole, and the resource control policy that cuts outside AssumeRole did not exist until 03:20 EDT. The attacker was back in as an administrator, session `ops`, 14 minutes after the SCP went on, and the SCP blocked only what that session tried at 01:39 EDT: `DeleteDBInstance`. Earlier, at 01:31 EDT, a `DeleteImportedKeyMaterial` on the data-lake key was denied too (`EV-00566`). It came from the second key's user `aws-support-sync` inside `wf-prod-data`, the key's own account, because KMS allows that call only there. It is the attacker's lock-now move: deleting the material makes everything under the key unreadable at once, and only the attacker holds the material that could undo it.

Selected rows after the containment SCP, including every denied row from then on. The second key was active from 01:31 to 03:01. Times are MM-DD HH:MM:SS; account names drop the prefix `wf-`.

Row	Local (EDT)	UTC	Account	Principal and session	API call and result	Evidence
AT-151	07-04 01:23:10	07-04 05:23:10	`prod-lending`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00538`
AT-154	07-04 01:24:05	07-04 05:24:05	`prod-data`	`svc-artifact-sync`	`AssumeRole` on `role/wf-deploy-pipeline`	`EV-00542`
AT-155	07-04 01:24:35	07-04 05:24:35	`prod-data`	`jenkins-sync-4550`	`CreateAccessKey` on `aws-support-sync`; Denied: AccessDenied	`EV-00544`
AT-156	07-04 01:31:14	07-04 05:31:14	`prod-data`	`aws-support-sync`	`DeleteImportedKeyMaterial` on `key/33767b29-2aaa-5253-8…`; Denied: AccessDeniedException	`EV-00566`
AT-157	07-04 01:36:52	07-04 05:36:52	`prod-lending`	`s-9172` (exfil-runner, account `777788889999`)	`AssumeRole` on `role/OrgAccessBackup`	`EV-00569`
AT-158	07-04 01:39:05	07-04 05:39:05	`prod-lending`	`ops` (OrgAccessBackup)	`DeleteDBInstance` on `wf-los-prod`; Denied: AccessDenied	`EV-00570`
AT-159	07-04 02:47:12	07-04 06:47:12	`prod-data`	`aws-support-sync`	`GetCallerIdentity`	`EV-00637`
AT-162	07-04 02:49:31	07-04 06:49:31	`prod-data`	`aws-support-sync`	`GetSessionToken`	`EV-00640`
AT-164	07-04 02:59:30	07-04 06:59:30	`prod-data`	`aws-support-sync`	`ListUsers` (2 calls, to 03:00); Denied: AccessDenied	`EV-00643` to `EV-00644`

Attribution

The evidence is consistent with a financially motivated actor. Three cues are also consistent with a Russian-speaking one, and our confidence in that is low. We track the actor as `VTL-C1`, we name no group, and we claim no country or person. `VTL-C1` is Vantalect's own tracking label, invented for this sample, and it names a cluster of tooling and infrastructure, not a person or a state. Financial motivation follows from the demand ($14,500,000 in cryptocurrency, with a deadline) and from the absence of any other goal we can see. No payment was made. The table states, for each cue, what in the record it rests on.

The cues behind the attribution wording, with the basis for each. None is conclusive alone, and none is a finding of who or where.

Cue	Basis in the record	How much it is worth
Working hours. The discovery calls fell between 02:00 and 11:00 EDT (06:00 to 15:00 UTC) on each of four days, June 29 to July 2.	The trail (rows AT-001 to AT-052). On each day the first call was at or after 02:00 EDT and the last at or before 10:51 EDT. The destructive work began on a US holiday evening.	Low. The window is consistent with a working day three hours ahead of UTC, a band that includes Moscow and many other places, and working hours are easy to copy. The holiday timing points to knowledge of the US calendar, not to a location.
Russian-language artifacts in the metadata of the ransom note.	Our reading of the note's object metadata (`EVD-004` holds the object and its metadata). The note's text and contact are withheld from this sample, so a reader of this report cannot check the reading.	Low to moderate. Metadata shows the settings of whoever prepared the file, and it can be planted. We ran no linguistic analysis.
Tooling and VPS infrastructure that overlap the cluster we track as VTL-C1.	The Python client string, the name imitation, and the hosting addresses, compared with our own earlier tracking. VTL-C1 is a cluster we defined from the same kinds of observable.	Moderate, and circular in part: the overlap shows that a toolset repeats, not who uses it. VTL-C1 is Vantalect's invented tracking label for the sample, not a named group.

Whether any sanctions screening applies, and what to screen, is for counsel to decide; we give counsel the indicators and make no determination. The wording above is not a basis to act on, and no one should act on it without counsel. The ransom contact is withheld from this report.

Dwell and the intervals that matter

The attacker had 4 days, 17 hours, 50 minutes between first use and the first write, and the bank had 4 hours 18 minutes between the first write and the control that stopped it. The shorter interval is the one a faster response could have cut. The longer one is the one that better detection could have cut: nothing in four days of read-only calls raised an alert.

The intervals between the attacker's steps and the bank's. Times are from the organization trail and the incident record.

Interval	From	To	Length
Dwell	June 29, 03:14 EDT (first use)	July 3, 21:04 EDT (first write)	4 days, 17 hours, 50 minutes
First write to the GuardDuty finding	July 3, 21:04 EDT	July 3, 22:44 EDT	1 hour 40 minutes
Finding to the page	July 3, 22:44 EDT	July 3, 23:18 EDT	34 minutes
Page to the emergency activation	July 3, 23:18 EDT	July 4, 00:52 EDT	1 hour 34 minutes
Activation to the containment SCP	July 4, 00:52 EDT	July 4, 01:22 EDT	30 minutes
First write to the containment SCP	July 3, 21:04 EDT	July 4, 01:22 EDT	4 hours 18 minutes

Open questions

The report leaves 7 questions open. For the first three, nothing in the bank's AWS logs can answer them. The first matters most for counsel: **object reads from the imaging archive were not logged**, so we cannot say whether any were made. We found no evidence of reads. That is not the same as evidence that there were none.

The open questions, in the order counsel is likely to ask them.

Question	Why it is open	What would answer it
Did the attacker read objects from the imaging archive?	S3 data events were logged for writes only, so a read leaves no record in any log we hold. We found no evidence of reads, which is not evidence that there were none.	Nothing in the bank's AWS records can answer it. Only material from the attacker's side (obtained through law enforcement) could. Until then the register carries it as possible and accepted (CI-031).
Was the database snapshot copied out?	The 7 KMS calls from account `777788889999` on the prod-lending key (a re-encryption to a key in that account, and a grant naming the database service with the attacker's role as retiring principal) are the pattern of a snapshot copy, and the share was open for 137 minutes. The copy itself happens in the attacker's account and is not in the bank's trail, so we cannot see whether it finished or what it moved.	Records from the attacker's account, which only law enforcement or AWS can reach, would confirm it. Until then: probable, with high confidence.
What else was on the contractor's laptop, and who else had the file?	The credentials file was taken by an infostealer (counsel's interview). It may have held other secrets. This is outside AWS and outside our evidence.	A forensic image of the laptop, which counsel would have to arrange with the contractor. We can search AWS for any other key from the same contractor once counsel supplies the names.
Was June 29 the first use of the key?	It is the first use we can see in the logs we hold. The key dates from March 14, 2023. The week-two search of the sealed trail history for June 1 to June 28 (EVD-025) found no use of the key and no call from the three outside addresses, so the question is open only for the time before June 1.	A search of the longest-retained logs before June 1 would test it; the estate map (report forensics-1) lists what retention allows. Counsel's interview puts the infostealer on June 22.
What did GuardDuty and the provider's queue hold before 22:44, and how was each item triaged?	The incident record gives one finding, the 22:44 one, and a page 34 minutes later. We hold no ticket or runbook entry from the provider. GuardDuty documents finding types that the earlier activity would be expected to raise (see the assumptions note in the detection section); we have not seen any in the record.	The provider's queue and ticket history for June 29 to July 4, and the export in EVD-005 read finding by finding. The bank can ask the provider for both.

Question	Why it is open	What would answer it
Does the attacker still hold any access?	Every identity the attacker created is deleted, the outside trust (OrgAccessBackup) was deleted on July 8. The three attacker keys, whose material expired at 18:00 on July 7, were kept as they are for counsel: they have been in PendingImport since then, the outside statement was dropped from each key policy on July 4 at 02:09, and the bank's `DisableKey` call on each key on July 8 at 10:03 to 10:04 was rejected with `KMSInvalidStateException` (EV-01052, EV-01054, EV-01056), because a key in PendingImport cannot be disabled. We cannot prove a negative from the logs.	The week-four closure sweep repeats the week-one map and every indicator search, and the seven-day watch covers the days after the changes. Neither has been done as of this report.
Did the attacker act in any account we have not listed?	Nothing in the organization trail points to one. An account outside the organization would not appear in the trail.	The indicators in the companion report, matched against any other log the bank keeps (VPN, proxy, identity provider), by the bank or its provider.

What happens next

The register of everything the attacker created, changed, or touched is in the companion report, Compromise Inventory, with the status of each item as of July 24. The root causes and the weaknesses elsewhere in the estate are in Root Cause and Vulnerable Infrastructure. The Investigation's readout to counsel and the sponsor is today.

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.

Appendix A: the complete timeline

Every one of the 164 rows of the attacker's activity, in time order. Times are 2026, month-day and hours:minutes:seconds; local time is EDT, and UTC is beside it. The evidence column gives the event numbers in Vantalect's copy of the organization trail. A row stands for one call, or for a run of the same call by the same principal in one burst (the count and the end time are in the API column). The session named `s-9172` is the `exfil-runner` role in account `777788889999`; `ops` is the `OrgAccessBackup` role.

The complete attacker timeline, 164 rows, from June 29, 03:14 EDT to Saturday, July 4, 02:59 EDT.

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-001	06-29 03:14:02	06-29 07:14:02	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`GetCallerIdentity`	Allowed	`EV-00003`
AT-002	06-29 03:20:27	06-29 07:20:27	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`ListAccounts`	Denied: AccessDeniedException	`EV-00004`
AT-003	06-29 03:47:37	06-29 07:47:37	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`ListAliases`	Allowed	`EV-00005`
AT-004	06-29 04:22:22	06-29 08:22:22	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`GetRole` on `wf-jenkins-agent`	Allowed	`EV-00006`
AT-005	06-29 06:06:50	06-29 10:06:50	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`ListKeys`	Allowed	`EV-00007`
AT-006	06-29 06:20:16	06-29 10:20:16	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`DescribeDBInstances`	Allowed	`EV-00008`
AT-007	06-29 07:08:39	06-29 11:08:39	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`ListAttachedUserPolicies` on `svc-artifact-sync`	Allowed	`EV-00009`
AT-008	06-29 07:23:46	06-29 11:23:46	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`ListBuckets`	Allowed	`EV-00010`
AT-009	06-29 07:26:09	06-29 11:26:09	`wf-tools`	`svc-artifact-sync`	`198.51.100.23`	`GetRole` on `wf-artifact-sync-task`	Allowed	`EV-00011`

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-010	06-29 08:00:24	06-29 12:00:24	wf-tools	svc-artifact-sync	198.51.100.23	ListRoles	Allowed	EV-00012
AT-011	06-29 09:36:09	06-29 13:36:09	wf-tools	svc-artifact-sync	198.51.100.23	ListAccessKeys on svc-artifact-sync	Allowed	EV-00013
AT-012	06-29 10:51:07	06-29 14:51:07	wf-tools	svc-artifact-sync	198.51.100.23	ListRoles	Allowed	EV-00014
AT-013	06-30 02:03:00	06-30 06:03:00	wf-tools	svc-artifact-sync	198.51.100.23	GetCallerIdentity	Allowed	EV-00017
AT-014	06-30 03:42:52	06-30 07:42:52	wf-tools	svc-artifact-sync	198.51.100.23	ListAccessKeys on svc-artifact-sync	Allowed	EV-00018
AT-015	06-30 03:52:34	06-30 07:52:34	wf-tools	svc-artifact-sync	198.51.100.23	DescribeDBInstances	Allowed	EV-00019
AT-016	06-30 04:02:03	06-30 08:02:03	wf-tools	svc-artifact-sync	198.51.100.23	ListRoles	Allowed	EV-00020
AT-017	06-30 04:08:57	06-30 08:08:57	wf-tools	svc-artifact-sync	198.51.100.23	ListAccounts	Denied: AccessDeniedException	EV-00021
AT-018	06-30 04:13:36	06-30 08:13:36	wf-tools	svc-artifact-sync	198.51.100.23	ListAttachedUserPolicies on svc-artifact-sync	Allowed	EV-00022
AT-019	06-30 04:20:14	06-30 08:20:14	wf-tools	svc-artifact-sync	198.51.100.23	ListRoles	Allowed	EV-00023
AT-020	06-30 05:25:07	06-30 09:25:07	wf-tools	svc-artifact-sync	198.51.100.23	ListAliases	Allowed	EV-00024
AT-021	06-30 07:06:14	06-30 11:06:14	wf-tools	svc-artifact-sync	198.51.100.23	GetRole on wf-artifact-sync-task	Allowed	EV-00025
AT-022	06-30 07:29:04	06-30 11:29:04	wf-tools	svc-artifact-sync	198.51.100.23	GetRole on wf-jenkins-agent	Allowed	EV-00026

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-023	06-30 08:24:27	06-30 12:24:27	wf-tools	svc-artifact-sync	198.51.100.23	ListKeys	Allowed	EV-00027
AT-024	06-30 09:51:06	06-30 13:51:06	wf-tools	svc-artifact-sync	198.51.100.23	ListBuckets	Allowed	EV-00028
AT-025	07-01 02:00:00	07-01 06:00:00	wf-tools	svc-artifact-sync	198.51.100.41	GetCallerIdentity	Allowed	EV-00033
AT-026	07-01 02:08:00	07-01 06:08:00	wf-tools	svc-artifact-sync	198.51.100.41	ListAccounts	Denied: AccessDeniedException	EV-00036
AT-027	07-01 02:51:56	07-01 06:51:56	wf-tools	svc-artifact-sync	198.51.100.41	ListRoles	Allowed	EV-00037
AT-028	07-01 04:08:45	07-01 08:08:45	wf-tools	svc-artifact-sync	198.51.100.41	GetRole on wf-jenkins-agent	Allowed	EV-00038
AT-029	07-01 04:17:15	07-01 08:17:15	wf-tools	svc-artifact-sync	198.51.100.41	DescribeDBInstances	Allowed	EV-00039
AT-030	07-01 05:16:58	07-01 09:16:58	wf-tools	svc-artifact-sync	198.51.100.41	ListKeys	Allowed	EV-00040
AT-031	07-01 06:44:42	07-01 10:44:42	wf-tools	svc-artifact-sync	198.51.100.41	ListRoles	Allowed	EV-00041
AT-032	07-01 07:27:05	07-01 11:27:05	wf-tools	svc-artifact-sync	198.51.100.41	ListAliases	Allowed	EV-00042
AT-033	07-01 07:45:53	07-01 11:45:53	wf-tools	svc-artifact-sync	198.51.100.41	ListAttachedUserPolicies on svc-artifact-sync	Allowed	EV-00043
AT-034	07-01 07:51:03	07-01 11:51:03	wf-tools	svc-artifact-sync	198.51.100.41	GetRole on wf-artifact-sync-task	Allowed	EV-00044
AT-035	07-01 08:49:14	07-01 12:49:14	wf-tools	svc-artifact-sync	198.51.100.41	ListAccessKeys on svc-artifact-sync	Allowed	EV-00045

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-036	07-01 10:00:16	07-01 14:00:16	wf-tools	svc-artifact-sync	198.51.100.41	ListBuckets	Allowed	EV-00050
AT-037	07-01 10:21:14	07-01 14:21:14	wf-nonprod-data	svc-artifact-sync	198.51.100.41	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00051
AT-038	07-01 10:22:05	07-01 14:22:05	wf-nonprod-data	jenkins-sync-4440	198.51.100.41	ListBuckets	Allowed	EV-00052
AT-039	07-01 10:22:30	07-01 14:22:30	wf-nonprod-data	jenkins-sync-4440	198.51.100.41	DescribeDBInstances	Allowed	EV-00053
AT-040	07-01 10:22:55	07-01 14:22:55	wf-nonprod-data	jenkins-sync-4440	198.51.100.41	ListKeys	Allowed	EV-00054
AT-041	07-02 02:04:00	07-02 06:04:00	wf-tools	svc-artifact-sync	198.51.100.41	GetCallerIdentity	Allowed	EV-00057
AT-042	07-02 02:07:11	07-02 06:07:11	wf-tools	svc-artifact-sync	198.51.100.41	GetRole on wf-artifact-sync-task	Allowed	EV-00058
AT-043	07-02 04:24:39	07-02 08:24:39	wf-tools	svc-artifact-sync	198.51.100.41	ListBuckets	Allowed	EV-00059
AT-044	07-02 05:28:36	07-02 09:28:36	wf-tools	svc-artifact-sync	198.51.100.41	ListAttachedUserPolicies on svc-artifact-sync	Allowed	EV-00060
AT-045	07-02 05:48:01	07-02 09:48:01	wf-tools	svc-artifact-sync	198.51.100.41	ListRoles	Allowed	EV-00061
AT-046	07-02 06:16:14	07-02 10:16:14	wf-tools	svc-artifact-sync	198.51.100.41	DescribeDBInstances	Allowed	EV-00062
AT-047	07-02 06:17:26	07-02 10:17:26	wf-tools	svc-artifact-sync	198.51.100.41	ListRoles	Allowed	EV-00063
AT-048	07-02 06:32:38	07-02 10:32:38	wf-tools	svc-artifact-sync	198.51.100.41	ListAccounts	Denied: AccessDeniedException	EV-00064

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-049	07-02 06:54:22	07-02 10:54:22	wf-tools	svc-artifact-sync	198.51.100.41	GetRole on wf-jenkins-agent	Allowed	EV-00065
AT-050	07-02 07:12:32	07-02 11:12:32	wf-tools	svc-artifact-sync	198.51.100.41	ListKeys	Allowed	EV-00066
AT-051	07-02 10:06:04	07-02 14:06:04	wf-tools	svc-artifact-sync	198.51.100.41	ListAliases	Allowed	EV-00067
AT-052	07-02 10:17:34	07-02 14:17:34	wf-tools	svc-artifact-sync	198.51.100.41	ListAccessKeys on svc-artifact-sync	Allowed	EV-00068
AT-053	07-03 21:04:11	07-04 01:04:11	wf-prod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00083
AT-054	07-03 21:06:08	07-04 01:06:08	wf-prod-lending	jenkins-sync-4471	203.0.113.47	CreateKey on key/b9885f41-4ff8-5cbf-9…	Allowed	EV-00084
AT-055	07-03 21:06:15	07-04 01:06:15	wf-prod-lending	jenkins-sync-4471	203.0.113.47	CreateAlias on alias/wf-backup-2026	Allowed	EV-00085
AT-056	07-03 21:06:27	07-04 01:06:27	wf-prod-lending	jenkins-sync-4471	203.0.113.47	GetParametersForImport on b9885f41-4ff8-5cbf-9f54-…	Allowed	EV-00086
AT-057	07-03 21:07:00	07-04 01:07:00	wf-prod-lending	jenkins-sync-4471	203.0.113.47	ImportKeyMaterial on b9885f41-4ff8-5cbf-9f54-…	Allowed	EV-00087
AT-058	07-03 21:09:02	07-04 01:09:02	wf-prod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00088
AT-059	07-03 21:09:21	07-04 01:09:21	wf-prod-imaging	jenkins-sync-4473	203.0.113.47	CreateKey on key/6d9bd1af-48a0-5150-b…	Allowed	EV-00089

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-060	07-03 21:09:28	07-04 01:09:28	wf-prod-imaging	jenkins-sync-4473	203.0.113.47	CreateAlias on alias/wf-backup-2026	Allowed	EV-00090
AT-061	07-03 21:09:40	07-04 01:09:40	wf-prod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00091
AT-062	07-03 21:09:40	07-04 01:09:40	wf-prod-imaging	jenkins-sync-4473	203.0.113.47	GetParametersForImport on 6d9bd1af-48a0-5150-b619-…	Allowed	EV-00092
AT-063	07-03 21:09:55	07-04 01:09:55	wf-prod-data	jenkins-sync-4475	203.0.113.47	CreateKey on key/33767b29-2aaa-5253-8…	Allowed	EV-00093
AT-064	07-03 21:10:02	07-04 01:10:02	wf-prod-data	jenkins-sync-4475	203.0.113.47	CreateAlias on alias/wf-backup-2026	Allowed	EV-00094
AT-065	07-03 21:10:13	07-04 01:10:13	wf-prod-imaging	jenkins-sync-4473	203.0.113.47	ImportKeyMaterial on 6d9bd1af-48a0-5150-b619-…	Allowed	EV-00095
AT-066	07-03 21:10:14	07-04 01:10:14	wf-prod-data	jenkins-sync-4475	203.0.113.47	GetParametersForImport on 33767b29-2aaa-5253-83bf-…	Allowed	EV-00096
AT-067	07-03 21:10:47	07-04 01:10:47	wf-prod-data	jenkins-sync-4475	203.0.113.47	ImportKeyMaterial on 33767b29-2aaa-5253-83bf-…	Allowed	EV-00097
AT-068	07-03 21:12:03	07-04 01:12:03	wf-prod-data	jenkins-sync-4475	203.0.113.47	CopyObject (214 calls, to 21:12) on wf-datalake-prod	Denied: AccessDenied	EV-00098 to EV-00311
AT-069	07-03 21:14:26	07-04 01:14:26	wf-prod-imaging	jenkins-sync-4473	203.0.113.47	PutBucketLifecycleConfiguration on wf-imaging-archive-prod	Allowed	EV-00312

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-070	07-03 21:20:05	07-04 01:20:05	wf-prod-digital	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00313
AT-071	07-03 21:20:45	07-04 01:20:45	wf-prod-digital	jenkins-sync-4478	203.0.113.47	ListBuckets	Allowed	EV-00314
AT-072	07-03 21:21:05	07-04 01:21:05	wf-prod-digital	jenkins-sync-4478	203.0.113.47	DescribeDBInstances	Allowed	EV-00315
AT-073	07-03 21:21:25	07-04 01:21:25	wf-prod-digital	jenkins-sync-4478	203.0.113.47	ListKeys	Allowed	EV-00316
AT-074	07-03 21:22:17	07-04 01:22:17	wf-prod-lending	jenkins-sync-4471	203.0.113.47	CreateDBSnapshot on wf-los-prod	Allowed	EV-00317
AT-075	07-03 21:22:30	07-04 01:22:30	wf-prod-treasury	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00318
AT-076	07-03 21:23:10	07-04 01:23:10	wf-prod-treasury	jenkins-sync-4480	203.0.113.47	ListBuckets	Allowed	EV-00319
AT-077	07-03 21:23:30	07-04 01:23:30	wf-prod-treasury	jenkins-sync-4480	203.0.113.47	DescribeDBInstances	Allowed	EV-00320
AT-078	07-03 21:23:50	07-04 01:23:50	wf-prod-treasury	jenkins-sync-4480	203.0.113.47	ListKeys	Allowed	EV-00321
AT-079	07-03 21:26:00	07-04 01:26:00	wf-nonprod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00322
AT-080	07-03 21:26:40	07-04 01:26:40	wf-nonprod-lending	jenkins-sync-4482	203.0.113.47	ListBuckets	Allowed	EV-00323
AT-081	07-03 21:27:00	07-04 01:27:00	wf-nonprod-lending	jenkins-sync-4482	203.0.113.47	DescribeDBInstances	Allowed	EV-00324
AT-082	07-03 21:27:10	07-04 01:27:10	wf-nonprod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00325

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-083	07-03 21:27:20	07-04 01:27:20	wf-nonprod-lending	jenkins-sync-4482	203.0.113.47	ListKeys	Allowed	EV-00326
AT-084	07-03 21:27:50	07-04 01:27:50	wf-nonprod-imaging	jenkins-sync-4483	203.0.113.47	ListBuckets	Allowed	EV-00327
AT-085	07-03 21:28:00	07-04 01:28:00	wf-nonprod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00328
AT-086	07-03 21:28:10	07-04 01:28:10	wf-nonprod-imaging	jenkins-sync-4483	203.0.113.47	DescribeDBInstances	Allowed	EV-00329
AT-087	07-03 21:28:30	07-04 01:28:30	wf-nonprod-imaging	jenkins-sync-4483	203.0.113.47	ListKeys	Allowed	EV-00330
AT-088	07-03 21:28:40	07-04 01:28:40	wf-nonprod-data	jenkins-sync-4485	203.0.113.47	ListBuckets	Allowed	EV-00331
AT-089	07-03 21:29:00	07-04 01:29:00	wf-nonprod-digital	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00332
AT-090	07-03 21:29:00	07-04 01:29:00	wf-nonprod-data	jenkins-sync-4485	203.0.113.47	DescribeDBInstances	Allowed	EV-00333
AT-091	07-03 21:29:20	07-04 01:29:20	wf-nonprod-data	jenkins-sync-4485	203.0.113.47	ListKeys	Allowed	EV-00334
AT-092	07-03 21:29:40	07-04 01:29:40	wf-nonprod-digital	jenkins-sync-4488	203.0.113.47	ListBuckets	Allowed	EV-00335
AT-093	07-03 21:30:00	07-04 01:30:00	wf-nonprod-treasury	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00336
AT-094	07-03 21:30:00	07-04 01:30:00	wf-nonprod-digital	jenkins-sync-4488	203.0.113.47	DescribeDBInstances	Allowed	EV-00337
AT-095	07-03 21:30:20	07-04 01:30:20	wf-nonprod-digital	jenkins-sync-4488	203.0.113.47	ListKeys	Allowed	EV-00338

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-096	07-03 21:30:40	07-04 01:30:40	wf-nonprod-treasury	jenkins-sync-4491	203.0.113.47	ListBuckets	Allowed	EV-00339
AT-097	07-03 21:31:00	07-04 01:31:00	wf-nonprod-treasury	jenkins-sync-4491	203.0.113.47	DescribeDBInstances	Allowed	EV-00340
AT-098	07-03 21:31:20	07-04 01:31:20	wf-nonprod-treasury	jenkins-sync-4491	203.0.113.47	ListKeys	Allowed	EV-00341
AT-099	07-03 21:34:10	07-04 01:34:10	wf-prod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00342
AT-100	07-03 21:34:20	07-04 01:34:20	wf-prod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00343
AT-101	07-03 21:35:10	07-04 01:35:10	wf-prod-lending	jenkins-sync-4494	203.0.113.47	CreateSnapshot (31 calls, to 21:36) on snapshot/snap-0d8ae7cbff…	Allowed	EV-00344 to EV-00389
AT-102	07-03 21:35:22	07-04 01:35:22	wf-prod-data	jenkins-sync-4496	203.0.113.47	CreateSnapshot (15 calls, to 21:35) on snapshot/snap-09f30351bb…	Allowed	EV-00351 to EV-00379
AT-103	07-03 21:38:50	07-04 01:38:50	wf-prod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00390
AT-104	07-03 21:39:12	07-04 01:39:12	wf-prod-imaging	jenkins-sync-4499	203.0.113.47	CreateRole on s3-batch-replication	Allowed	EV-00391
AT-105	07-03 21:40:02	07-04 01:40:02	wf-prod-imaging	jenkins-sync-4499	203.0.113.47	PutRolePolicy on s3-batch-replication	Allowed	EV-00392
AT-106	07-03 21:41:00	07-04 01:41:00	wf-prod-imaging	jenkins-sync-4499	203.0.113.47	CreateJob on role/s3-batch-replication	Allowed	EV-00393
AT-107	07-03 21:41:05	07-04 01:41:05	wf-prod-imaging	AWSService (S3 Batch Operations)	batchoperations.s3.amazonaws.com	AssumeRole on role/s3-batch-replication	Allowed	EV-00394

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-108	07-03 21:47:00	07-04 01:47:00	wf-prod-lending	jenkins-sync-4494	203.0.113.47	CopySnapshot (31 calls, to 21:48) on snapshot/snap-0fbdf84542…	Allowed	EV-00395 to EV-00440
AT-109	07-03 21:47:01	07-04 01:47:01	wf-prod-data	jenkins-sync-4496	203.0.113.47	CopySnapshot (15 calls, to 21:47) on snapshot/snap-091f899d61…	Allowed	EV-00396 to EV-00424
AT-110	07-03 21:56:30	07-04 01:56:30	wf-prod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00441
AT-111	07-03 21:58:31	07-04 01:58:31	wf-prod-lending	jenkins-sync-4502	203.0.113.47	CopyDBSnapshot on wf-los-prod-bk-20260703	Allowed	EV-00442
AT-112	07-03 22:03:58	07-04 02:03:58	wf-prod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00443
AT-113	07-03 22:05:03	07-04 02:05:03	wf-prod-lending	jenkins-sync-4505	203.0.113.47	DeleteDBSnapshot (9 calls, to 22:05) on wf-los-prod-monthend-202…	Allowed	EV-00444 to EV-00452
AT-114	07-03 22:08:50	07-04 02:08:50	wf-prod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00453
AT-115	07-03 22:09:04	07-04 02:09:04	wf-prod-lending	jenkins-sync-4505	203.0.113.47	DeleteSnapshot (31 calls, to 22:09) on snap-02b6db40772885b3f	Allowed	EV-00454 to EV-00485
AT-116	07-03 22:09:33	07-04 02:09:33	wf-prod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00483
AT-117	07-03 22:09:40	07-04 02:09:40	wf-prod-data	jenkins-sync-4510	203.0.113.47	DeleteSnapshot (15 calls, to 22:09) on snap-0e136e7b45205b0d1	Allowed	EV-00486 to EV-00500
AT-118	07-03 22:14:18	07-04 02:14:18	wf-prod-lending	jenkins-sync-4505	203.0.113.47	DeleteTrail on wf-lending-legacy-trail	Allowed	EV-00501

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-119	07-03 22:15:04	07-04 02:15:04	wf-prod-lending	jenkins-sync-4505	203.0.113.47	StopLogging on trail/wf-org-trail	Denied: AccessDeniedException	EV-00502
AT-120	07-03 22:19:58	07-04 02:19:58	wf-prod-digital	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00503
AT-121	07-03 22:22:20	07-04 02:22:20	wf-prod-treasury	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00504
AT-122	07-03 22:25:50	07-04 02:25:50	wf-nonprod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00505
AT-123	07-03 22:27:05	07-04 02:27:05	wf-nonprod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00506
AT-124	07-03 22:27:55	07-04 02:27:55	wf-nonprod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00507
AT-125	07-03 22:28:50	07-04 02:28:50	wf-nonprod-digital	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00508
AT-126	07-03 22:29:55	07-04 02:29:55	wf-nonprod-treasury	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00509
AT-127	07-03 22:31:06	07-04 02:31:06	wf-prod-data	jenkins-sync-4510	203.0.113.47	CreateUser on aws-support-sync	Allowed	EV-00510
AT-128	07-03 22:31:31	07-04 02:31:31	wf-prod-data	jenkins-sync-4510	203.0.113.47	AttachUserPolicy on aws-support-sync	Allowed	EV-00511
AT-129	07-03 22:31:58	07-04 02:31:58	wf-prod-data	jenkins-sync-4510	203.0.113.47	CreateAccessKey on aws-support-sync	Allowed	EV-00512
AT-130	07-03 22:36:14	07-04 02:36:14	wf-prod-lending	jenkins-sync-4505	203.0.113.47	CreateRole on OrgAccessBackup	Allowed	EV-00513

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-131	07-03 22:36:40	07-04 02:36:40	wf-prod-lending	jenkins-sync-4505	203.0.113.47	AttachRolePolicy on OrgAccessBackup	Allowed	EV-00514
AT-132	07-03 23:04:20	07-04 03:04:20	wf-prod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00515
AT-133	07-03 23:09:12	07-04 03:09:12	wf-prod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00516
AT-134	07-03 23:09:41	07-04 03:09:41	wf-prod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00517
AT-135	07-03 23:30:11	07-04 03:30:11	wf-prod-imaging	jenkins-sync-4530	203.0.113.47	PutObject on wf-imaging-archive-prod	Allowed	EV-00518
AT-136	07-03 23:30:36	07-04 03:30:36	wf-prod-data	jenkins-sync-4532	203.0.113.47	PutObject on wf-datalake-prod	Allowed	EV-00519
AT-137	07-03 23:51:12	07-04 03:51:12	wf-prod-lending	jenkins-sync-4527	203.0.113.47	ModifyDBSnapshotAttribute on wf-los-prod-bk-20260703	Allowed	EV-00520
AT-138	07-04 00:04:07	07-04 04:04:07	wf-prod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00521
AT-139	07-04 00:06:09	07-04 04:06:09	wf-prod-lending	s-9172 (exfil-runner, account 777788889999)	203.0.113.47	CreateGrant (2 calls, to 00:06) on key/b9885f41-4ff8-5cbf-9…	Allowed	EV-00522 to EV-00524
AT-140	07-04 00:06:21	07-04 04:06:21	wf-prod-lending	s-9172 (exfil-runner, account 777788889999)	203.0.113.47	DescribeKey on key/b9885f41-4ff8-5cbf-9…	Allowed	EV-00523
AT-141	07-04 00:07:02	07-04 04:07:02	wf-prod-lending	s-9172 (exfil-runner, account 777788889999)	rds.amazonaws.com	ReEncrypt on key/b9885f41-4ff8-5cbf-9…	Allowed	EV-00525
AT-142	07-04 00:08:59	07-04 04:08:59	wf-prod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00526

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-143	07-04 00:09:25	07-04 04:09:25	wf-prod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00527
AT-144	07-04 00:28:14	07-04 04:28:14	wf-prod-lending	s-9172 (exfil-runner, account 777788889999)	rds.amazonaws.com	Decrypt on key/b9885f41-4ff8-5cbf-9…	Allowed	EV-00528
AT-145	07-04 00:47:09	07-04 04:47:09	wf-prod-lending	jenkins-sync-4534	203.0.113.47	ModifyDBInstance on wf-los-prod	Allowed	EV-00529
AT-146	07-04 00:50:48	07-04 04:50:48	wf-prod-lending	s-9172 (exfil-runner, account 777788889999)	rds.amazonaws.com	Decrypt on key/b9885f41-4ff8-5cbf-9…	Allowed	EV-00530
AT-147	07-04 00:51:06	07-04 04:51:06	wf-prod-lending	s-9172 (exfil-runner, account 777788889999)	203.0.113.47	RetireGrant on key/b9885f41-4ff8-5cbf-9…	Allowed	EV-00531
AT-148	07-04 01:04:15	07-04 05:04:15	wf-prod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00532
AT-149	07-04 01:09:03	07-04 05:09:03	wf-prod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00533
AT-150	07-04 01:09:30	07-04 05:09:30	wf-prod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00534
AT-151	07-04 01:23:10	07-04 05:23:10	wf-prod-lending	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00538
AT-152	07-04 01:23:40	07-04 05:23:40	wf-prod-imaging	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00540
AT-153	07-04 01:23:40	07-04 05:23:40	wf-prod-lending	jenkins-sync-4547	203.0.113.47	GetCallerIdentity	Allowed	EV-00541
AT-154	07-04 01:24:05	07-04 05:24:05	wf-prod-data	svc-artifact-sync	203.0.113.47	AssumeRole on role/wf-deploy-pipeline	Allowed	EV-00542
AT-155	07-04 01:24:35	07-04 05:24:35	wf-prod-data	jenkins-sync-4550	203.0.113.47	CreateAccessKey on aws-support-sync	Denied: AccessDenied	EV-00544

Row	Local (EDT)	UTC	Account	Principal and session	Source	API call	Result	Evidence
AT-156	07-04 01:31:14	07-04 05:31:14	wf-prod-data	aws-support-sync	203.0.113.47	DeleteImportedKeyMaterial on key/33767b29-2aaa-5253-8…	Denied: AccessDeniedException	EV-00566
AT-157	07-04 01:36:52	07-04 05:36:52	wf-prod-lending	s-9172 (exfil-runner, account 777788889999)	203.0.113.47	AssumeRole on role/OrgAccessBackup	Allowed	EV-00569
AT-158	07-04 01:39:05	07-04 05:39:05	wf-prod-lending	ops (OrgAccessBackup)	203.0.113.47	DeleteDBInstance on wf-los-prod	Denied: AccessDenied	EV-00570
AT-159	07-04 02:47:12	07-04 06:47:12	wf-prod-data	aws-support-sync	203.0.113.47	GetCallerIdentity	Allowed	EV-00637
AT-160	07-04 02:47:52	07-04 06:47:52	wf-prod-data	aws-support-sync	203.0.113.47	ListAccessKeys on aws-support-sync	Allowed	EV-00638
AT-161	07-04 02:48:32	07-04 06:48:32	wf-prod-data	aws-support-sync	203.0.113.47	ListBuckets	Allowed	EV-00639
AT-162	07-04 02:49:31	07-04 06:49:31	wf-prod-data	aws-support-sync	203.0.113.47	GetSessionToken	Allowed	EV-00640
AT-163	07-04 02:50:10	07-04 06:50:10	wf-prod-data	GetSessionToken session (aws-support-sync, temporary session)	203.0.113.47	DescribeDBInstances	Allowed	EV-00641
AT-164	07-04 02:59:30	07-04 06:59:30	wf-prod-data	aws-support-sync	203.0.113.47	ListUsers (2 calls, to 03:00)	Denied: AccessDenied	EV-00643 to EV-00644

© 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.