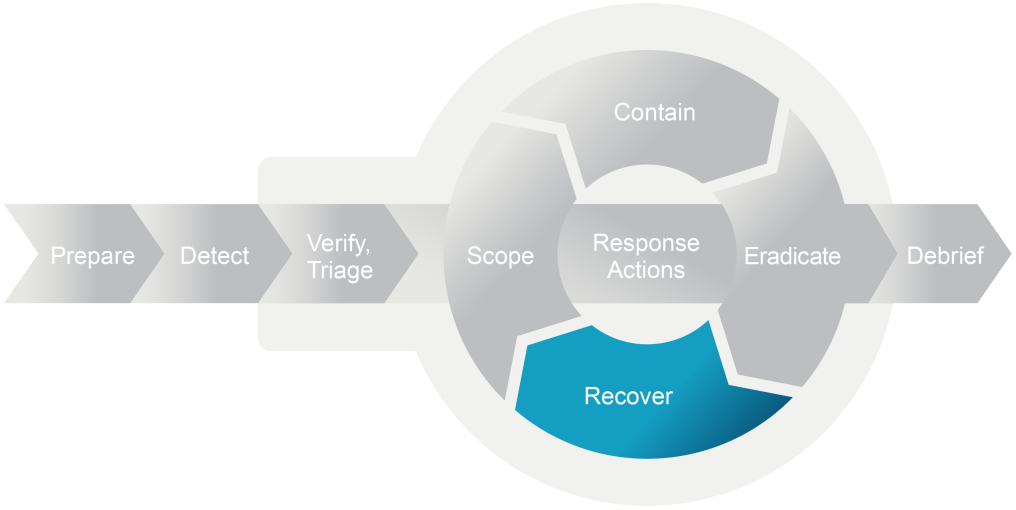
Recover: Step-by-Step



The following steps provide a condensed reference for recovery activities. Each step corresponds to topics covered in the Recover Activity chapter of *Dynamic Incident Response* (www.dynamicincidentresponse.com), organized for use when validating, testing, and coordinating the return of systems to production.

Step 1. Conduct Pre-Restoration Verification

This step revalidates the eradicate Step 8 exit criteria for each system about to return to production rather than repeating the eradication work from scratch. Use the evidence and documentation produced during eradication as the starting point, and focus Step 1 on confirming those findings still hold and on the verification activities that are specific to recovery, such as backup integrity and rebuild integrity, which are not part of eradication.

1. Revalidate root cause remediation before restoration, including:
    - Confirm that the exploited vulnerability remains patched on the system.
    - Verify compromised credentials are still rotated across all systems where they were used.
    - Validate the misconfigurations enabling initial access remain corrected.
    - Review eradication documentation to confirm all identified issues were addressed and no regressions have been introduced since eradicate Step 8.
2. Revalidate persistence mechanism removal, including:
    - Treat eradication verification as an ongoing activity rather than a single decision.
    - Confirm the eradicate Step 8 persistence checks still hold on the system about to be restored.
    - Spot-check scheduled tasks, services, registry entries, and unauthorized accounts for any reappearance since the eradicate sign-off.
    - Perform registry and configuration validation, scheduled task review, startup item inspection, and integrity checks for critical binaries, feeding the results into go/no-go decisions for each recovery step.
3. Validate backup integrity for backup-based restorations, including:
    - Confirm backup date predates the initial compromise based on the incident timeline.
    - Verify backup integrity through checksum validation or test restoration.
    - Document backup selection rationale for incident records.

    - If no clean backup exists, document the rebuild approach and validation steps.
4. Verify rebuild integrity for rebuilt systems, including:
    - Confirm rebuild used trusted installation media or gold images.
    - Verify installation sources have not been compromised.
    - Document the rebuild process and any deviations from standard procedures.

Step 2. Perform System Validation Testing

1. Conduct functional testing, including:
    - Run standard operational tests to verify core business functions.
    - Test data access, transaction processing, and application workflows.
    - Use existing test plans or UAT documentation where available.
    - Document any functional issues discovered during testing.
2. Verify security control configuration, including:
    - Confirm the EDR agent is installed, running, and reporting to the management console.
    - Verify the host-based firewall is enabled and configured per organizational policy.
    - Check that logging is enabled and events are flowing to collection systems.
    - Validate patches and hardening applied during eradication remain in place.
3. Perform inter-connectivity testing, including:
    - Test connections to dependent databases and verify data access.
    - Verify communication paths to other application servers.
    - Test authentication flows for users and service accounts.
    - Confirm network connectivity to required internal and external resources.

Step 3. Obtain System Owner Acceptance

1. Coordinate acceptance testing with system owners, including:
    - Provide test plans or validation checklists to system owners.
    - Schedule acceptance testing window with business unit representatives.
    - Support system owners during their validation activities.
    - Document any issues identified during owner acceptance testing.
2. Document acceptance and obtain sign-off, including:
    - Record what testing was performed and by whom.
    - Capture the owner's acknowledgment that the system is ready for production.
    - Document any known limitations or issues accepted by the owner.
    - Obtain formal sign-off before proceeding to production restoration.

Step 4. Configure Enhanced Monitoring

1. Enable elevated logging on restored systems, including:
    - Configure advanced audit policies for authentication, process creation, and system changes.
    - Enable PowerShell Script Block Logging on Windows systems.
    - Configure auditd rules for critical file and process monitoring on Linux systems.
    - Verify logs are flowing to SIEM or log collection infrastructure.

2. Enable and validate incident-specific detection rules on restored systems, including:
    - Confirm the detection rules authored during contain and eradicate (command-and-control domains, file hashes, behavioral patterns based on observed attacker TTPs) are active and targeted at the restored systems.
    - Tune alert thresholds and scope for the enhanced monitoring window if signal-to-noise needs adjustment now that the systems are back in production.
    - Add any rules identified during recovery scoping that were not implemented earlier, coordinating with detection engineering rather than creating ad hoc rules.
    - Test detection rules to confirm they generate expected alerts on the restored systems.
3. Establish monitoring duration and procedures, including:
    - Define a monitoring period appropriate to incident severity (typically thirty days minimum).
    - Document what constitutes abnormal behavior requiring investigation.
    - Establish response procedures for alerts on recovered systems.
    - Assign responsibility for monitoring review during the enhanced monitoring period.

Step 5. Execute Coordinated Production Restoration

1. Identify system dependencies and restoration sequence, including:
    - Map dependencies between affected systems.
    - Identify foundational services (AD, DNS, network infrastructure) requiring early restoration.
    - Document the restoration sequence based on dependency analysis.
    - Coordinate sequence with system owners and IT operations teams.
2. Coordinate restoration timing with stakeholders, including:
    - Present scheduling options and associated risks to decision-makers, recommending off-hours restoration where feasible to reduce user impact, ease monitoring, and provide rollback flexibility.
    - Recognize that timing decisions ultimately belong to organizational leadership.
    - Document guidance provided and decisions made regarding timing, including rationale when leadership chooses immediate restoration against recommendations.
    - Communicate the restoration schedule to all affected teams.
    - Prepare rollback procedures in case restoration encounters problems.
3. Select phased or coordinated restoration based on context, including:
    - Use phased recovery when eradication confidence is limited or capacity is constrained. Restore a small set of systems first, enable monitoring, and watch for anomalies before scaling up.
    - Use coordinated recovery for interconnected systems with well-documented dependencies that require simultaneous restoration.
    - Restore systems according to the planned sequence.
    - Validate each system or phase before proceeding to the next.
    - Monitor for issues during and immediately after each restoration.
    - Document any deviations from the planned restoration sequence.
4. Manage recovery coordination complexity, including:
    - Designate an incident response coordinator for incidents spanning multiple systems and teams to track progress, facilitate communication, and ensure steps execute in the proper sequence.
    - Define explicit handoff criteria between restoration, validation, acceptance testing, and production release.
    - Translate technical recovery work into business language when engaging with leadership to manage business pressure for rapid restoration.

- Prepare communication templates for common scenarios (delays, partial restoration, user action required) before recovery begins.
- Establish post-recovery baseline documentation through thorough acceptance testing and owner sign-off to defend against misplaced problem attribution when unrelated issues emerge later.

5. Address cloud recovery considerations (when applicable), including:
   - Verify cloud snapshot creation dates against the incident timeline before restoring instances.
   - Audit IAM access mechanisms, including API keys, access tokens, service account credentials, role assignments, and policies.
   - Validate that multi-factor authentication is enabled for all users with access to cloud resources.
   - Review infrastructure-as-code templates and version control history for unauthorized modifications.
   - Consider increasing the verbosity of cloud logging (API access, network connections, resource changes) during post-recovery monitoring.
6. Coordinate user communications, including:
   - Designate communication leads to manage messaging to affected users and stakeholders.
   - Establish a regular communication cadence with stakeholders and stick to it, even when there is no new information.
   - Avoid overpromising specific restoration timelines, as delays may erode credibility.
   - Prepare template messages for common scenarios (delays, partial restoration, user action required).
   - Use multiple communication channels, recognizing that some systems (email, chat) may be unavailable.
7. Address data loss from clean-backup recovery, including:
   - Identify data created or modified since the last clean backup.
   - Recover user-created files and application data from newer backups only after verifying they are free of attacker artifacts.
   - Coordinate with business owners on acceptable data loss thresholds.
   - Document data recovery decisions and any accepted losses.

Step 6. Remove Containment Measures

Containment removal for individual systems occurs incrementally during Step 5 as each system completes validation and returns to production; a fully isolated system cannot be restored or tested in place without selectively lifting the measures that block its operation. Step 6 consolidates that work by inventorying all measures implemented during contain, tracking their status as each system returns, and evaluating which measures should be retained as durable controls rather than removed when recovery closes.

1. Inventory containment measures in place, including:
   - Document all firewall rules, network segmentation, and access restrictions.
   - Record disabled accounts, blocked services, and DNS sinkholes.
   - Note the rationale and implementation date for each measure.
   - Identify measures that should remain in place as permanent improvements.
2. Remove containment measures incrementally, including:
   - Remove measures affecting each system as it completes validation and returns to production.
   - Monitor for adverse effects after each removal of a containment measure.
   - Document each removal action with a timestamp and the outcome.
   - Verify system functionality after containment measures are removed.

3. Evaluate containment measures for permanent implementation, including:
    - Assess which temporary measures provide long-term security value.
    - Work with security architecture teams on permanent implementation.
    - Document decisions about which measures to keep versus remove.
    - Update security policies to reflect any permanent changes.

Step 7. Capture Recovery Metrics

1. Record recovery timeline for each system, including:
    - Document when restoration began for each affected system.
    - Record validation completion, owner acceptance, and production restoration timestamps.
    - Calculate total recovery duration and identify any bottlenecks.
    - Compare the actual timeline against any estimates provided to stakeholders.
2. Document issues and resolutions, including:
    - Record problems encountered during restoration and how they were resolved.
    - Document any adaptations to planned recovery procedures.
    - Capture lessons learned while the information is fresh.
    - Note any gaps in recovery documentation or procedures discovered during execution.
3. Track resource investment, including:
    - Record personnel hours spent on recovery activities.
    - Document third-party support services engaged and their contributions.
    - Capture any additional costs incurred during recovery.
    - Provide resource data for incident cost analysis and future planning.
4. Pass recovery documentation to debrief for consolidation, including:
    - Hand off the recovery documentation in the form expected by debrief Step 3 (which consolidates documentation across all phases into the unified incident record).
    - Ensure documentation is accessible for post-incident review.
    - Prepare a summary handoff for ongoing enhanced monitoring (Step 4) and for the AAR.