Amazon-Scale Threat Modeling

Overview

This guide covers Amazon's threat modeling interview approach, focusing on systematic analysis at massive scale with customer trust as the primary concern.

Amazon's Threat Modeling Interview Format

Typical Question

> "Our customer service team needs a file upload feature where customers can submit support documents. This will serve 100 million customers globally. Walk me through how you'd threat model this system."

What Amazon Evaluates (5 Key Areas)

1. **Systematic Approach** - Do you follow a clear methodology (STRIDE, PASTA)?
2. **Scale Considerations** - Can you think at Amazon's 100M+ user scale?
3. **Customer Impact Focus** - How do security threats affect customer trust?
4. **Business Risk Prioritization** - What threats matter most to the business?
5. **Scalable Solutions** - Will your mitigations work at Amazon scale?

The Amazon Threat Modeling Framework

Step 1: System Understanding (2-3 minutes)

Essential Clarifying Questions:

- "What types of files will customers upload?" → *Determines data sensitivity*
- "What's the expected upload volume daily?" → *Scale planning*
- "Who can access uploaded files?" → *Access control scope*
- "Are there compliance requirements?" → *Regulatory constraints*
- "What's the business criticality level?" → *Risk tolerance*

Sample Response:

> "Before starting the threat model, let me understand scope. For 100M global customers uploading support documents, we're handling potentially sensitive personal information across AWS regions. Any security incident would significantly impact customer trust given our scale."

Step 2: Amazon-Scale Architecture (3-4 minutes)

Architecture for 100M Users:

```
[Customer Browser] → [CloudFront CDN] → [ALB] → [API Gateway] → [Lambda/ECS]
                                                                      ↓
[S3 Multi-Bucket] ← [KMS Encryption] ← [DLP Scanning] ← [Virus Scanning]
```

```
                                                                    ↓
[CloudTrail] → [Security Hub] → [GuardDuty] → [Automated Response]
```

Key AWS Services Integration:

- **CloudFront**: Global delivery + DDoS protection
- **WAF**: Application-level filtering
- **S3**: Scalable storage with versioning
- **KMS**: Centralized key management
- **Lambda**: Serverless processing for scale
- **GuardDuty**: ML-powered threat detection
- **Macie**: Data classification and DLP

Step 3: STRIDE Analysis with Amazon Context (8-10 minutes)

S - Spoofing Identity

Threat: Attackers impersonating customers to access/upload files

Amazon Scale Impact:

- 100M users = credential stuffing attacks highly likely
- 0.1% compromise = 100K affected customers

Customer Impact:

- Unauthorized access to personal support documents
- Loss of trust in Amazon's security
- Customer churn and reputation damage

Mitigations:

- AWS Cognito with MFA enforcement
- Device fingerprinting and risk analysis
- Rate limiting per IP/user/region
- Behavioral analytics for anomaly detection

Business Justification:

> "At 100M users, even 0.1% account compromise affects 100K customers. Customer trust impact could cost $50M+ in churn based on industry data."

T - Tampering with Data

Threat: Malicious file uploads (malware, oversized files, content manipulation)

Amazon Scale Impact:

- Could affect infrastructure serving all customers
- Malware distribution through trusted platform
- Resource exhaustion across global infrastructure

Customer Impact:

- Service degradation for legitimate users
- Potential malware exposure
- Loss of platform trust

Mitigations:

- Multi-layer scanning (ClamAV + ML detection)
- File type validation with deep content inspection
- Size limits with overflow to Glacier
- Sandboxed processing in isolated Lambda environments
- Content integrity verification with checksums

Business Justification:

> "One malware incident could trigger regulatory action affecting entire customer base. Prevention cost: $2M/year vs. incident cost: $500M+ in fines and reputation damage."

R - Repudiation

Threat: Customers/attackers denying file upload actions

Amazon Scale Impact:

- Investigation costs multiply across millions of transactions
- Legal liability for disputed evidence
- Audit complexity across global infrastructure

Customer Impact:

- Disputes over support case evidence
- Delayed resolution of customer issues
- Trust in platform integrity

Mitigations:

- Comprehensive CloudTrail logging with immutable storage
- Digital signatures on all uploads
- Chain of custody documentation
- Tamper-evident audit logs in separate AWS account

I - Information Disclosure

Threat: Unauthorized access to customer files

Amazon Scale Impact:

- Massive data breach potential (100M+ records)
- Global regulatory compliance violations
- Cross-region data exposure

Customer Impact:

- Personal information exposure
- Privacy violations and identity theft risk
- Complete loss of platform trust

Mitigations:

- Encryption at rest (KMS) and in transit (TLS 1.3)
- Least privilege access with IAM roles
- Data classification and automated handling
- Regional data residency compliance
- Access logging with real-time monitoring

Business Justification:

> "100M customer records at $165/record (IBM 2023) = $16.5B potential breach cost. GDPR fines up to 4% of global revenue."

D - Denial of Service

Threat: Resource exhaustion through large/numerous uploads

Amazon Scale Impact:

- Could affect global customer service capability
- Infrastructure costs spike unexpectedly
- Cascading failures across services

Customer Impact:

- Unable to submit support requests
- Degraded service performance
- Frustration with platform reliability

Mitigations:

- CloudFront caching and DDoS protection
- API Gateway throttling with burst handling
- S3 request rate optimization
- Auto-scaling with cost controls
- Circuit breaker patterns

E - Elevation of Privilege

Threat: Attackers gaining admin access through file uploads

Amazon Scale Impact:

- Could compromise entire customer service infrastructure
- Access to global customer database
- Platform-wide security breach

Customer Impact:

- Mass data exposure across all customers
- Complete service outages
- Permanent loss of trust

Mitigations:

- Sandboxed file processing in containers
- Fargate for isolated execution
- IAM roles with minimal permissions
- Network segmentation with VPCs
- Runtime security monitoring

Step 4: Risk Prioritization (2-3 minutes)

Critical Priority (Fix Immediately):

1. **Information Disclosure** - Encryption, access controls, data residency
2. **Tampering** - Malware scanning, content validation, sandboxing

High Priority (Fix This Sprint): 3. **Denial of Service** - Rate limiting, resource management, scaling 4. **Elevation of Privilege** - Container security, IAM hardening

Medium Priority (Next Release): 5. **Spoofing** - Advanced authentication, behavioral analytics 6. **Repudiation** - Enhanced audit capabilities, digital signatures

Step 5: Amazon-Scale Implementation (2-3 minutes)

Scalable Solution Principles:

```
# Example: Amazon-scale file validation service
class AmazonFileValidator:
    def __init__(self):
        self.s3_client = boto3.client('s3')
        self.sqs_client = boto3.client('sqs')

    def validate_upload_async(self, bucket, key, customer_id):
        """Async validation to handle 100M+ users"""

        # Immediate basic validation (< 100ms)
        metadata = self.s3_client.head_object(Bucket=bucket, Key=key)
        if metadata['ContentLength'] > MAX_FILE_SIZE:
            return self.reject_upload("File exceeds limit")

        # Queue comprehensive scanning (doesn't block customer)
        validation_job = {
            'bucket': bucket,
            'key': key,
            'customer_id': customer_id,
            'timestamp': time.time()
        }
```

```
        self.sqs_client.send_message(
            QueueUrl=VALIDATION_QUEUE_URL,
            MessageBody=json.dumps(validation_job)
        )

        # Customer gets immediate confirmation
        return self.notify_customer_upload_received(customer_id)
```

Amazon Interview Response Framework

Winning Response Structure (15-20 minutes total):

1. **System Understanding** (2-3 min): Ask clarifying questions, show business thinking
2. **Architecture Design** (3-4 min): Draw AWS-native, scalable architecture
3. **STRIDE Analysis** (8-10 min): Systematic threat analysis with customer impact
4. **Risk Prioritization** (2-3 min): Business-driven threat ranking
5. **Scalable Solutions** (2-3 min): AWS implementation approach

Sample Complete Interview Response:

Clarification (2 minutes):

> "Let me understand the business context first. With 100M customers uploading support documents globally, we're handling sensitive customer data with high availability requirements and regulatory compliance across multiple jurisdictions. The business impact of any security incident would be severe given our customer trust requirements."

Architecture (3 minutes):

> "I'll design this using AWS services optimized for scale: CloudFront for global distribution and DDoS protection, API Gateway for request management and throttling, Lambda for serverless processing that auto-scales, S3 for durable storage with KMS encryption, and Security Hub for centralized monitoring. This architecture handles millions of uploads daily with automatic scaling and built-in security."

STRIDE Analysis (10 minutes):

> "Starting with Spoofing threats - at 100M users, we face significant credential stuffing risk. Even 0.1% compromise affects 100K customers, potentially costing $50M in customer churn based on industry averages..."
>
> "For Tampering, malicious uploads pose the highest business risk. One malware incident could trigger regulatory action affecting our entire platform. I recommend multi-layer scanning with immediate containment..."
>
> "Information Disclosure is our top priority - 100M customer records represent $16.5B in potential breach costs and GDPR fines up to 4% of revenue..."

Risk Prioritization (2 minutes):

> "My immediate priorities are Information Disclosure and Tampering - these directly threaten customer trust and could trigger company-threatening incidents. DoS and privilege escalation are high priority but manageable with proper AWS configuration..."

Implementation (2 minutes):

> "For Amazon's scale, I recommend async processing architecture where customers receive immediate upload confirmation, then background services perform comprehensive security scanning without blocking user experience. This maintains customer satisfaction while ensuring thorough security validation."

Amazon-Specific Success Factors

Customer Obsession Integration

- **Always start with customer impact**: "How does this threat affect our customers?"
- **Frame security as enabler**: "This protection allows customers to trust us with sensitive data"
- **Consider global diversity**: "Different regions have different threat profiles and compliance needs"

Scale Considerations

- **Think in millions**: Solutions must work for 100M+ users
- **Consider peak loads**: Black Friday, Prime Day traffic spikes
- **Global distribution**: Multi-region, multi-AZ architectures
- **Cost at scale**: $1 per user = $100M total cost

Business Communication

- **Quantify everything**: Specific dollar amounts for risk and mitigation costs
- **Use industry benchmarks**: IBM breach costs, Ponemon studies
- **Compare alternatives**: Build vs. buy decisions with clear ROI
- **Timeline impact**: Customer trust takes years to build, seconds to lose

This systematic approach demonstrates the methodical thinking, scale awareness, and customer focus that Amazon values in security engineers.