Dell Unisphere Mock API

A Comprehensive Solution for Software Upgrade Testing

The Problem

- Development and testing of Unity storage system software upgrades requires access to Dell Unisphere API
- Actual Unity storage hardware is:
  - Expensive
  - Limited availability
  - Difficult to use in CI/CD pipelines
  - Hard to reproduce specific test scenarios
- Need a consistent, controllable environment for testing various upgrade scenarios

Requirements Overview

- Implement a realistic mock of Dell Unisphere API
- Support full software upgrade workflow
- Match REST API structure and response formats
- Implement authentication and security features
- Support resource operations (GET, POST, PUT, DELETE)
- Handle file uploads for software packages
- Simulate asynchronous upgrade processes

Main API Components

- Authentication and session management
- Basic system information
- User management
- Software version management
- Upgrade session workflow
- Task status monitoring and simulation

Key Technical Challenges

- Maintaining stateful sessions
- Simulating asynchronous operations
- Implementing realistic upgrade workflows with proper task transitions
- Ensuring API response formats match real Dell Unisphere API
- Implementing security features (authentication, CSRF protection)
- Supporting concurrent requests
- Managing file uploads (10MB+ software packages)

Basic System Info Response


```
{
    "@base": "http://localhost:8000/api/types/basicSystemInfo/instances?per_page=2000",
    "updated": "2025-03-15T14:06:42.076Z",
    "links": [
        {
            "rel": "self",
            "href": "&page=1"
        }
    ],
    "entries": [
        {
            "@base": "http://localhost:8000/api/instances/basicSystemInfo",
            "content": {
                "id": "0",
                "model": "Unity 380F",
                "name": "CKM01204905476",
                "softwareVersion": "5.3.0",
                "softwareFullVersion": "Unity 5.3.0.0 (Release, Build 120, 2023-03-18 19:02:01, 5.3.0.0.5.120)",
                "apiVersion": "13.0",
                "earliestApiVersion": "4.0"
            },
            "links": [
                {
                    "rel": "self",
                    "href": "/0"
                }
            ],
            "updated": "2025-03-15T14:06:42.076Z"
        }
    ]
}
```

Authentication Flow

Login Session Response


```
{
    "@base": "http://localhost:8000/api/instances/loginSessionInfo",
    "content": {
        "id": "admin",
        "roles": [
            {
                "id": "administrator"
            }
        ],
        "user": {
            "id": "user_admin"
        },
        "domain": "local",
        "idleTimeout": 3600,
        "isPasswordChangeRequired": false
    },
    "links": [
        {
            "rel": "self",
            "href": "/admin"
        }
```

User Resource Example


```
{
    "@base": "http://localhost:8000/api/types/user/instances?per_page=2000",
    "updated": "2025-03-15T14:06:42.335Z",
    "links": [
        {
            "rel": "self",
            "href": "&page=1"
        }
    ],
    "entries": [
        {
            "@base": "http://localhost:8000/api/instances/user",
            "content": {
                "id": "user_admin"
            },
            "links": [
                {
                    "rel": "self",
                    "href": "/0"
                }
            ],
            "updated": "2025-03-15T14:06:42.335Z"
        },
        {
            "@base": "http://localhost:8000/api/instances/user",
            "content": {
                "id": "user_user"
            },
            "links": [
                {
                    "rel": "self",
                    "href": "/1"
                }
            ],
            "updated": "2025-03-15T14:06:42.335Z"
        }
    ]
}
```

Installed Software Version Resource


```
{
    "@base": "http://localhost:8000/api/instances/installedSoftwareVersion",
    "content": {
        "id": "0",
        "version": "5.3.0",
        "revision": 120,
        "releaseDate": "2025-03-15T14:06:27.934947",
        "fullVersion": "Unity 5.3.0.0 (Release, Build 120, 2023-03-18 19:02:01, 5.3.0.0.5.120)",
        "languages": [
            {
                "name": "English",
                "version": "5.3.0"
            },
            {
                "name": "Chinese",
                "version": "5.3.0"
            }
        ],
        "hotFixes": [
            "HF1",
            "HF2"
        ],
        "packageVersions": [
            {
                "name": "Base",
                "version": "5.3.0"
            },
            {
                "name": "Management",
                "version": "5.3.0"
            }
        ]
    },
    "links": [
        {
            "rel": "self",
            "href": "/0"
        }
    ],
    "updated": "2025-03-15T14:06:42.479Z"
}
```

Software Upgrade Process Flow

1. Upload software package to `/upload/files/types/softwareUploadPackage`
2. Prepare candidate software version
3. Verify upgrade eligibility
4. Create upgrade session
5. Track upgrade progress through status polling
6. Resume upgrade if paused/interrupted
7. Verify successful completion

Step 1: Software Upload

Request

```
curl -s -k -L -X POST "http://localhost:8000/upload/files/types/candidateSoftwareVersion" \
        -u "admin:Password123!" \
        -b cookie.jar \
        -H "X-EMC-REST-CLIENT: true" \
        -H "EMC-CSRF-TOKEN: d7b466aa-bea6-47e6-8464-107adf22d77d" \
        -F "file=@./tests/scripts/test_results/test_upgrade.bin"
```

Response

```
{
    "id": "file_c04e72c9-bf87-4b08-a08c-890de1317d74",
    "filename": "test_upgrade.bin",
    "size": 10485760
}
```

Step 2: Prepare Software Package

Request

```
curl -s -k -L -X POST "http://localhost:8000/api/types/candidateSoftwareVersion/action/prepare" \
        -u "admin:Password123!" \
        -b cookie.jar \
        -H "X-EMC-REST-CLIENT: true" \
        -H "EMC-CSRF-TOKEN: d7b466aa-bea6-47e6-8464-107adf22d77d" \
        -H "Content-Type: application/json" \
        -d '{"filename":"file_c04e72c9-bf87-4b08-a08c-890de1317d74"}'
```

Response

```
{
    "id": "candidate_f285cc52-5a69-4fa5-b058-83dc2b3fd3b2",
    "status": "SUCCESS"
}
```

Step 3: Retrieve Candidate Software


```
{
    "@base": "http://localhost:8000/api/types/candidateSoftwareVersion/instances?per_page=2000",
    "updated": "2025-03-15T14:06:43.118Z",
    "entries": [
        {
            "@base": "http://localhost:8000/api/instances/candidateSoftwareVersion",
            "content": {
                "id": "candidate_f285cc52-5a69-4fa5-b058-83dc2b3fd3b2",
                "version": "5.4.0",
                "fullVersion": "Unity 5.4.0.0 (Release, Build 150, 2023-06-18 19:02:01, 5.4.0.0.5.150)",
                "revision": 150,
                "releaseDate": "2025-03-15T14:06:43.070722",
                "type": "SOFTWARE",
                "rebootRequired": true,
                "canPauseBeforeReboot": true
            },
            "links": [
                {
                    "rel": "self",
                    "href": "/candidate_f285cc52-5a69-4fa5-b058-83dc2b3fd3b2"
                }
            ],
            "updated": "2025-03-15T14:06:43.118Z"
        }
    ]
}
```

Step 4: Verify Upgrade Eligibility

Request

```
curl -s -k -L -X POST "http://localhost:8000/api/types/upgradeSession/action/verifyUpgradeEligibility" \
        -u "admin:Password123!" \
        -b cookie.jar \
        -H "X-EMC-REST-CLIENT: true" \
        -H "EMC-CSRF-TOKEN: 00b5dfa7-8cc6-49e5-bb46-5c9f475fd0c5"
```

Response


```
{
    "eligible": true,
    "messages": [],
    "requiredPatches": [],
    "requiredHotfixes": []
}
```

Step 5: Create Upgrade Session

Response Example (Initial State)


```
{
    "@base": "http://localhost:8000/api/instances/upgradeSession",
    "content": {
        "id": "Upgrade_5.4.0.0",
        "status": 1,
        "percentComplete": 0,
        "tasks": [
            {
                "status": 1,
                "type": 0,
                "caption": "Preparing system",
                "creationTime": "2025-03-15T14:06:43.239288",
                "estRemainTime": "00:03:30.000"
            },
            {
                "status": 0,
                "type": 0,
                "caption": "Performing health checks",
                "creationTime": "2025-03-15T14:06:43.239288",
                "estRemainTime": "00:02:10.000"
            }
            // Additional pending tasks
```

Step 6: Monitor Upgrade Progress (start)

Task State Changes at 14:06:43

Task	Status
Preparing system	IN_PROGRESS
Performing health checks	PENDING
Preparing system software	PENDING
Waiting for reboot command	PENDING
Performing health checks	PENDING
Installing new software on peer SP	PENDING
Rebooting peer SP	PENDING
Restarting services on peer SP	PENDING

Step 6: Monitor Upgrade Progress (end)

NOTE: We speed up the simulation for testing purposes by a factor of 120

Task State Changes at 14:07:31

Task	Status
Preparing system	COMPLETED
Performing health checks	COMPLETED
Preparing system software	COMPLETED
Waiting for reboot command	COMPLETED
Performing health checks	COMPLETED
Installing new software on peer SP	COMPLETED
Rebooting peer SP	COMPLETED

Final Upgrade Session State


```
{
    "@base": "http://localhost:8000/api/instances/upgradeSession",
    "content": {
        "id": "Upgrade_5.4.0.0",
        "status": 2,
        "percentComplete": 100,
        "tasks": [
            {
                "status": 2,
                "type": 0,
                "caption": "Preparing system",
                "creationTime": "2025-03-15T14:06:43.239288",
                "estRemainTime": "00:03:30.000"
            },
            // Additional completed tasks...
            {
                "status": 2,
                "type": 2,
                "caption": "Final tasks",
                "creationTime": "2025-03-15T14:06:43.239288",
                "estRemainTime": "00:00:45.000"
            }
        ]
    }
}
```

Test Results

API Functionality Tests

- Basic system information retrieval ✅
- Session management (login/logout) ✅
- User management operations ✅
- Resource collection endpoints ✅
- Resource instance endpoints ✅

Test Results

Header Validation Tests

- Required X-EMC-REST-CLIENT header ✅
- CSRF token validation ✅
- Session cookie validation ✅
- Authorization header processing ✅
- Error responses for missing headers ✅

Test Results

Software Upgrade Flow

- File upload (10MB test package) ✅
- Candidate software creation ✅
- Upgrade session creation ✅
- Upgrade eligibility verification ✅
- Task state transitions ✅
- Progress monitoring ✅
- Successful completion ✅

Benefits Delivered

- Repeatable and reliable testing environment
- Support for CI/CD pipelines
- Faster development cycles without hardware dependencies
- Ability to test edge cases and error conditions
- Realistic API simulation for client application testing
- Comprehensive documentation of Dell Unisphere API behavior

Future Improvements

- Support for more resource types
- Enhanced error simulation capabilities
- Variable timing simulation for different system loads
- Network condition simulation (latency, packet loss)
- Support for additional upgrade scenarios
- Web UI for manual testing and visualization

Questions?

Thank you for your attention!