



Rathon-SSO v4.3

Java 연동 매뉴얼

라톤테크 인증기술팀

- 개정이력 -

버전	내용	일자	작성자
1.0	초안 작성 완료	2019.07.04	조항훈
1.4	Rathon-SSO v4.0 기준 최종 수정 버전	2019.11.05	조항훈
1.5	Rathon-SSO v4.1 기준 문서 정보 수정	2020.01.13	조항훈
1.6	Rathon-SSO v4.2.150 기준 문서 정보 수정	2021.05.26	조영민
1.7	SSO 서버 상태 체크 기능 내용 추가	2021.05.28	조항훈
1.8	Rathon-SSO v4.3.24 기준 문서 정보 수정	2021.11.01	조영민

목 차

1 개요	5
2 용어 및 약어	5
3 요구사항	6
4 설치 구성도	7
5 디렉토리 구조	7
6 어플리케이션 설정	8
6.1 web.xml	8
6.2 rathon_sso_sp.properties	9
7 Log 설정	10
7.1 logback.xml	10
7.2 다른 Logger의 사용	11
8 호출 Class 정의	12
8.1 com.rathontech.sso.sp.agent.web.WebAgent	12
8.2 com.rathontech.sso.sp.config.Env	13
8.3 com.rathontech.sso.sp.IdpHealthChecker	13
9 SSO 서버 상태 체크	14
9.1 WebAgent.requestAuthentication()	14
9.2 WebAgent.requestAuthentication(redirect)	14
9.3 그 외	14
10 예제	15
10.1 SSO 로그인	15
10.2 SSO 로그아웃	16



11 삭제	16
12 문제해결	16
12.1 SSO 서버와의 연결 문제	16
12.2 등록되지 않는 시스템 문제	16
12.3 요청 데이터 변조 문제	17
12.4 요청 무한루프 문제	17
12.5 응답 값 소실 문제	17
12.6 Single Logout 문제	18

1 개요

본 문서는 Rathon-SSO v4.3의 Java 기반의 시스템에 SSO SP(Agent)의 설치 방법을 안내하기 위한 문서로서 Agent의 배포와 설정, 연동 테스트에 대해 다루는 연동 안내 문서이다.

Java Agent는 JDK 1.5, 1.8 버전의 2가지가 존재하며 의존성 라이브러리의 버전 차이와 JDK 빌드 버전의 차이 외에는 기능상 동일하므로 특이사항이 없는 경우 두 버전 공통으로 설명을 진행한다.

2 용어 및 약어

본 문서에서 사용하는 용어와 약어 대한 정의를 나열한다.

시스템과 SSO와의 연동을 위한 매뉴얼이므로 다른 매뉴얼과 일부 표기 명칭을 달리 한다.

✓ SSO (Single Sign-On)

- 통합 인증, 싱글 사인온 단일 계정 로그인, 단일 인증 등을 의미한다.
- 한 번의 인증 과정으로 여러 컴퓨터 상의 자원을 이용 가능하게 하는 인증 기능.
- 이후 문서에서 '**SSO**'로 표기한다.

✓ 라톤테크 통합 인증 솔루션 v4.3

- 라톤테크에서 판매, 제공하는 SSO 솔루션 프로그램의 한글 명칭
- 이후 문서에서 '**Rathon-SSO v4.3**'으로 표기한다.

✓ Rathon-SSO v4.3

- 라톤테크에서 판매, 제공하는 SSO 솔루션 프로그램의 영문 명칭
- 이후 문서에서 '**Rathon-SSO v4.3**'으로 표기한다.

✓ SP (Service Provider)

- 서비스 제공자 통합 에이전트
- SSO를 적용할 연동 대상 시스템에 배포하는 라이브러리의 명칭
- 이후 문서에서 '**Agent**'로 표기한다.

✓ 연동 시스템

- SSO 연동 대상 시스템
- 기존에 사용 또는 신규로 도입되는 업무시스템에 배포한 Agent를 적용하여 Rathon-SSO v4.3과 연동하는 대상 시스템을 가르킨다.
- 이후 문서에서 '**연동 시스템**'으로 표기한다.

3 요구사항

항목	내용	비고
JVM	JDK 1.5 또는 1.8 이상	Oracle JDK Open JDK
WAS	Tomcat, Weblogic, WebSphere 등 Servlet을 사용할 수 있는 어플리케이션 서버	Tomcat 기준 5이상
Network	WAS가 프록시 방식 연동인 경우, (Switch 장비, Web Server 등) X-Forwarded-For, X-Forwarded-Proto 등 Header로 IP, Protocol 정보로 전달 설정	권장사항 필수는 아니지만 미설정시 정상적인 처리가 불가능 할 수 있음
Common Library	apache commons-codec 1.3 이상 jsoup 1.11.1 이상	Logging 관련 라이브러리는 기재하지 않음 그 외 버전은 호환 여부 문의
JDK 1.8 Library	apache commons-lang3 3.3.2 이상 apache httpclient 4.5.3 이상 apache httpcore 4.4.6 이상	
JDK 1.5 Library	apache commons-lang3 3.1 이상 apache httpclient 4.3.6 이상 apache httpcore 4.3.3 이상 joda-time 2.10.1 이상	
하드웨어	위 사양을 충족하는 구동 환경, 그외 별도 요구사항 없음	

문 서 명	Rathon-SSO v4.3 Java 연동 매뉴얼
작 성 자	조영민
작 성 일	2021. 11. 01.

4 설치 구성도

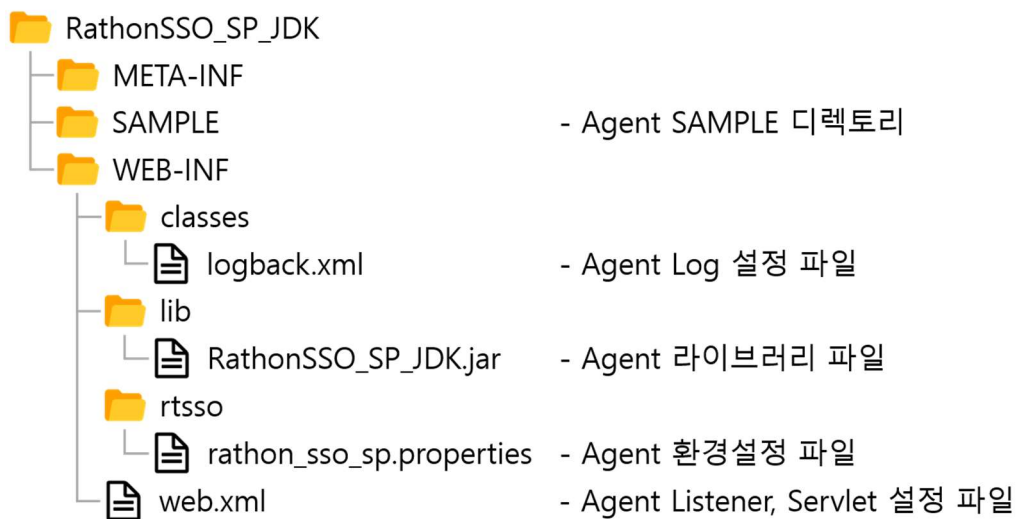
연동 시스템에 SSO Agent를 배포하고 설정한다.

연동 시스템
WAS



5 디렉토리 구조

배포파일은 war로 압축되어 있으며 OS에 관계없이 동일한 디렉토리 구조와 파일이 배포된다. 연동 Sample을 사용하기 위한 구조이며, 연동 시스템 환경에 맞게 설정 및 라이브러리를 복사하여 사용한다.



디렉토리명	설명	비고
SAMPLE	SSO와 Agent간의 연동 테스트를 위한 로그인, 페이지 프로텍션, 세션 확인 등의 각종 예제 페이지가 포함 되어있는 디렉토리	
classes	로그 설정 파일 저장 디렉토리	
lib	Agent를 빌드한 jar 패키지와 의존성 라이브러리들을 모아놓은 디렉토리	
rtssso	Agent 설정 파일 저장 디렉토리	
web.xml	Agent의 Listener, Servlet, Filter 등의 설정 정보를 저장	

6 어플리케이션 설정

설정은 web.xml과 rathon_sso_sp.properties 파일을 통해 설정한다.

```
## SSO Server URL
IDPM_DOMAIN_CONTEXT=http://10.30.89.1:8081/IDP

## Login Page URL
## Relative path: {contextPath}/SAMPLE/TestLogin.jsp
## Absolute path: https://localhost/SP/SAMPLE/TestLogin.jsp
SPO_LOGIN_PAGE_URI=/SAMPLE/TestLogin.jsp
```

(rathon_sso_sp.properties 설정 작성 예)

6.1 web.xml

파일 위치: {Web Application}/WEB-INF/web.xml

web.xml은 WAS에서 공통으로 사용되는 파일이므로 주요한 설정만 기술한다.

설정항목		설정 및 설명
filter		문자열 필터 설정 (옵션)
filter-name		com.rathontech.sso.sp.config.ServiceProviderCharsetFilter 기본값은 UTF-8이나 설정에 따라 달라진다. 기존에 사용하는 CharsetFilter가 존재하는 경우 문자열 인코딩이 문제가 되지 않는 범위에서 사용하지 않아도 무방하다.
context-param		context에 대한 parameter 설정 (필수)
param-name		RTSSO_SP_CONFIG parameter 이름 지정 수정하지 않는다.
	param-value	/WEB-INF/rtssso/rathon_sso_sp.properties 설정 파일의 위치 context-param을 통해 Agent에서 인식 가능한 위치라면 경로 및 파일명 수정이 가능하다.
listener		리스너 설정 (필수)
listener-class		com.rathontech.sso.sp.config.SPConfigurationListener Agent의 초기 기동에 필요한 Listener 클래스 지정, 수정하지 않는다.
servlet		Servlet 설정 (필수)
servlet-class		com.rathontech.sso.sp.agent.controller.ServiceProviderDispatcher SSO의 호출을 받는 Servlet 클래스 지정, 수정하지 않는다.
servlet-mapping		Servlet Mapping 설정 (필수)
url-pattern		/dispatch/* SSO의 호출을 받는 Servlet 주소 매핑, 변경시 rathon_sso_sp.properties 파일의 '*_RES_URI' 설정을 변경된 값에 맞춰 수정한다.

6.2 rathon_sso_sp.properties

파일 위치: {Web Application}/WEB-INF/rtssso/rathon_sso_sp.properties

web.xml 파일의 context-param/param-value 를 수정한 경우 해당 경로

설정항목	기본값	설명
IDPM_DOMAIN_CONTEXT		SSO 서버 주소 (절대경로)
SPO_LOGIN_PAGE_URI		SSO 로그인 페이지 주소 미설정시 SSO 기본 로그인 페이지를 사용한다. 상대경로/절대경로 모두 사용 가능
SPO_LOGIN_AFTER_PAGE		로그인 이후 강제 이동할 페이지 주소 설정시 Target Page를 무시하고 설정된 페이지로 이동한다. 상대경로/절대경로 모두 사용 가능
SPM_LOGOUT_RES_URI	/dispatch/slo	SSO Single Logout 호출 주소 web.xml의 servlet-mapping을 수정한 경우 맞게 수정한다. (상대경로)
SPM_SAML_RES_URI	/dispatch/sso	SSO Login 응답 주소 web.xml의 servlet-mapping을 수정한 경우 맞게 수정한다. (상대경로)
SPN_CLIENT_IP_HEADER	X-Forwarded-For	스위치 장비 등에 의해 포워드 되는 경우 전달하는 Client IP 저장 헤더 이름
SPN_CLIENT_PROTOCOL_HEADER	X-Forwarded-Proto	웹서버 등에 의한 프록시 환경인 경우 원호출 프로토콜의 값을 전달하는 헤더 이름
AGENT_PROVIDER_NAME		SSO 서버에 등록된 Agent의 시스템 ID 값 (관리자 문의)
AGENT_SIG_ALGORITHM		SSO 서버와 통신간 데이터 무결성 확인 알고리즘 HmacSHA256, HmacSHA384, HmacSHA512 (관리자 문의)
AGENT_SIG_ALG_KEY		SSO 서버와 통신간 데이터 무결성 확인 키(관리자 문의)
SPM_USE_IDP_AUTHENTICATION	true	SSO 인증 여부 false로 설정시 SSO 서버를 호출하지 않고 무조건 통과시킨다. 네트워크, SSO 서버 장애 등으로 인해 SSO가 불가능한 경우 사용한다.
IDPM_HEALTH_CHECK_USE	false	SSO 서버 상태 체크 사용 여부 Server To Server 통신이 가능해야 한다.
IDPM_HEALTH_CHECK_DELAY	PERIOD와 같은 값	SSO 서버 상태 체크 프로세스 시작 딜레이 시간 (단위: 초) 값이 비어있거나 0보다 작은 값인 경우 IDPM_HEALTH_CHECK_PERIOD와 같은 값으로 설정 된다.

문 서 명	Rathon-SSO v4.3 Java 연동 매뉴얼
작 성 자	조영민
작 성 일	2021. 11. 01.

설정항목	기본값	설명
IDPM_HEALTH_CHECK_PERIOD	60	SSO 서버 상태 체크 주기 (단위: 초) 설정된 시간마다 SSO 서버를 호출한다. SSO 서버의 부하를 고려하여 최저 설정 값은 10초이다.
IDPM_DOWN_REDIRECT_PAGE		SSO 서버 상태가 Down인 경우 이동할 대상 페이지 지정 상대경로/절대경로 모두 사용 가능

7 Log 설정

어플리케이션의 로그 기록은 Logback을 사용하고 있으며, 파일의 위치는 /WEB-INF/classes 아래 logback.xml 파일에 설정을 기록한다.

기존 logger와 충돌하는 경우 잘 조율하여 설정하도록 한다.

7.1 logback.xml

상위항목	항목	설명
property	LOG_PATTERN	출력 로그의 데이터 포맷 설정
	LOG_ENCODING	저장 파일의 인코딩 설정
	LOG_HISTORY	저장 파일의 유지 기간 일 단위
	LOG_SIZE	저장 파일의 크기 지정해 놓은 크기만큼 로그를 잘라서 저장함
	LOG_FILE	로그파일명 별다른 설정을 건드리지 않은 경우 {LOG_PATH}/RathonSSO_{LOG_FILE}.log 로 저장된다
	LOG_PATH	로그 저장 경로 지정 (OS따라 알맞은 경로를 작성한다)
appender	dailyRollingFileAppender	rollingPolicy/fileNamePattern에 의해 Rolling 된 파일은 {LOG_PATH}/archive/{LOG_FILE}/RathonSSO_{LOG_FILE}.{date}.{i}.log.zip으로 저장된다. '.zip'을 빼면 압축을 하지 않으며 '.tgz'로 방식을 변경 할 수 있다.
logger	{package name} or {target name}	필요와 상황에 따라 name란에 로깅할 패키지 또는 Target을 설정한다. logger는 여러 개 선언이 가능하며 각각 로그 레벨을 지정 할 수 있다. 기본은 com.rathontech의 디버거이며, 운영서버의 경우 성능을 위해 INFO 이상의 레벨을 사용하도록 설정한다.
root		※ 수정하지 않는다.

7.2 다른 Logger의 사용

기존 Logger가 Logback이 아닌 log4j, log4j2 등을 사용하는 하면서 Logback과의 충돌로 인해 Logback을 사용하지 않는 경우 참고한다.

설명은 log4j를 기준으로 하지만 각 Logger별로 설정이 크게 다르지 않다..

디렉토리	제거항목	추가항목
lib	jcl-over-slf4j.jar	log4j.jar
	logback-access.jar	slf4j-log4j12.jar
	logback-classic.jar	
	logback-core.jar	
classes	logback.xml	log4j.properties 또는 log4j.xml

```

4월 29, 2019 11:40:59 오전 org.apache.catalina.startup.HostConfig deployDirectory
정보: Deploying web application directory SP
2019-04-29 11:40:59,465 [main ] INFO SPCConfigurationListener - S
2019-04-29 11:40:59,467 [main ] DEBUG BaseConfigurationLoader - r
2019-04-29 11:40:59,469 [main ] DEBUG Env - 
2019-04-29 11:40:59,469 [main ] DEBUG Env - S
2019-04-29 11:40:59,470 [main ] DEBUG Env - 
2019-04-29 11:40:59,471 [main ] DEBUG Env - 
2019-04-29 11:40:59,471 [main ] DEBUG Env - 
2019-04-29 11:40:59,472 [main ] DEBUG Env - 
2019-04-29 11:40:59,472 [main ] DEBUG Env - 
2019-04-29 11:40:59,473 [main ] DEBUG Env - 
2019-04-29 11:40:59,473 [main ] DEBUG Env - 
2019-04-29 11:40:59,473 [main ] DEBUG Env - 
2019-04-29 11:40:59,474 [main ] DEBUG Env - 
2019-04-29 11:40:59,475 [main ] DEBUG Env - 
2019-04-29 11:40:59,476 [main ] DEBUG Env - 
2019-04-29 11:40:59,476 [main ] DEBUG Env - 
2019-04-29 11:40:59,476 [main ] DEBUG Env - 
2019-04-29 11:40:59,477 [main ] DEBUG Env - A
2019-04-29 11:40:59,477 [main ] DEBUG Env - A
  
```

(logback -> log4j Logger 교체)

8 호출 Class 정의

8.1 com.rathontech.sso.sp.agent.web.WebAgent

SSO 인증을 위해 필요한 Agent의 WebAgent 클래스에서 아래와 같은 기능을 제공한다.

Method	Input	Output	설명
requestAuthentication	HttpServletRequest HttpServletRequestes	boolean result -true: 로그인 필요 -false: 로그인 완료	사용자 인증 여부를 확인하기 위해 SSO 서버 호출하는 메소드. SSO 서버로의 redirect가 발생하며 로그인이 필요한 경우 로그인 페이지로 자동 redirect 한다.
requestAuthentication	HttpServletRequest HttpServletRequestes boolean redirect	boolean result -true: 로그인 필요 -false: 로그인 완료	사용자 인증 여부를 확인하기 위해 SSO 서버 호출하는 메소드. SSO 서버로의 redirect가 발생하며 로그인이 필요한 경우 로그인 페이지로 redirect 하지 않는다.
redirectLogout	HttpServletRequest HttpServletRequest	-	Redirect를 통한 로그아웃 방식 URL 파라미터 RelayState의 값에 정의된 주소로 로그아웃 후 Redirect 한다. 없으면 로그아웃 페이지에서 종료된다.
requestLogout	HttpServletRequest NULL	String result -S: 요청 성공 -F: 요청 실패	Redirect를 통한 로그아웃을 사용하지 못하는 환경에서 로그아웃 하는 방법 S 또는 F를 반환하지만 통신에 대한 성공여부이기 때문에 Logout 성공여부는 아니다.
getAuthentication	HttpServletRequest	Map<String, String> result -사용자 세션 정보	시스템에 SSO 세션이 생성이 되어있는 것을 필수 전제로 한다. 넥사크로등의 특수한 시스템 환경에서 requestAuthentication를 통하여 최초 SSO를 통한 세션 생성은 되나 사용중 갱신이 불가능한 경우 등에 SSO 서버로 직접 요청하여 사용자 세션 정보를 받아온다.
getLoginFormParameter	HttpServletRequest	String	Login 페이지에 SSO 연동을 하는 경우 SSO에 필요한 URL 파라미터를 받아 input 태그 문자열을 만들어 준다.

문 서 명	Rathon-SSO v4.3 Java 연동 매뉴얼
작 성 자	조영민
작 성 일	2021. 11. 01.

Method	Input	Output	설명
generateApiKey	HttpServletRequest	String	시스템에 SSO 세션이 생성이 되어있는 것을 필수 전제로 한다. verifyApiKey로 검증 가능한 ApiKey를 생성한다.
verifyApiKey	String (ApiKey)	Boolean - true: 유효 - false: 유효하지 않음.	generateApiKey로 생성된 KEY가 유효 한지 검증하는 메소드.
updateSession	HttpServletRequest	Boolean - true : 성공 - false : 실패	시스템에 SSO 세션이 생성이 되어있는 것을 필수 전제로 한다. 통합 인증 세션 및 연동 시스템의 SSO 세션을 갱신한다.

8.2 com.rathontech.sso.sp.config.Env

Type	Variable	
String	IDP_SESSION_KEY	SSO 세션 기본 키 값 (IDP_SESSION_ID)
String	DEFAULT_SESSION_USERID	사용자ID 세션 기본 키 값 (RathonSSO_USER_ID)
String	SSO_LOGIN_URL	SSO 로그인 주소
String	SSO_LOGOUT_URL	SSO 로그아웃 주소

8.3 com.rathontech.sso.sp.IdpHealthChecker

Method	Input	Output	설명
isAlive (static)		Boolean - true : Running - false : Down	SSO 서버 상태 체크 결과를 반환한다.

9 SSO 서버 상태 체크

6.2 `rathon_sso_sp.properties` 항목 설정과 `IdpHealthChecker` 클래스를 통해 SSO 서버의 상태를 확인할 수 있으며 연동 방식 등에 따라 적절한 구현이 필요하므로 상태 체크의 동작 방식과 Agent 기능에 미치는 영향에 대해 설명한다.

9.1 IdpHealthChecker의 동작

- IDPM_HEALTH_CHECK_USE 설정이 true 인 경우 IDPM_HEALTH_CHECK_DELAY 설정 시간과 관계없이 1 회 강제 호출하여 SSO 서버의 상태를 확인한다.
- IDPM_HEALTH_CHECK_PERIOD 설정에 따라 반복 실행되며 SSO 서버의 상태가 변하더라도 실행주기 또는 실행 여부가 변하지 않는다.
- SPM_USE_IDP_AUTHENTICATION 설정과는 개별 취급된다.
- SSO 서버와 연동 시스템 간의 Server To Server 통신이 가능해야 한다.

9.2 WebAgent.requestAuthentication()

- 자동으로 `response.sendRedirect` 를 호출하는 방식이다.
- Agent 를 통해 연동 시스템에 SSO 세션 정보가 생성되어있는 경우 SSO 서버 상태 체크 결과가 false 이더라도 IDPM_DOWN_REDIRECT_PAGE 설정 페이지를 호출하지 않는다.
- `IdpHealthChecker.isAlive()`가 true 이면, SPO_LOGIN_PAGE_URI 설정 페이지로 이동한다.
- `IdpHealthChecker.isAlive()`가 false 이면, IDPM_DOWN_REDIRECT_PAGE 설정 페이지로 이동한다.
- IDPM_DOWN_REDIRECT_PAGE 페이지는 연동 시스템에 따라 알맞게 구성한다.

9.3 WebAgent.requestAuthentication(redirect)

- redirect flag 값을 부여하여 `response.sendRedirect` 를 호출하는 방식이다.
- redirect 의 값이 true 인 경우, 9.2 항목을 참고한다.
- redirect 의 값이 false 인 경우, `response.sendRedirect` 를 호출하지 않기 때문에 직접 `IdpHealthChecker.isAlive()`를 호출하여 반환값에 따라 동작하는 로직을 구현한다.

9.4 그 외

- 연동 시스템 내 필요한 영역에 `IdpHealthChecker.isAlive()`를 사용하는 로직을 구현한다.



문 서 명	Rathon-SSO v4.3 Java 연동 매뉴얼
작 성 자	조영민
작 성 일	2021. 11. 01.

10 예제

배포한 SAMPLE 디렉토리에 있는 파일을 통한 SSO 로그인 프로세스만을 설명하며 Agent의 사용 방법에 대해서는 각 SAMPLE 페이지의 소스와 주석을 참고한다.

10.1 SSO 로그인

- 1) SAMPLE/TestSso.jsp (SSO 적용 페이지) 접근
- 2) WebAgent 에 의한 {SSO Server}/IDP/dispatch 호출 - 인증 여부 체크
- 3) SAMPLE/TestLogin.jsp 로그인 페이지로 리다이렉트됨.

Java Service Provider

ID

PW

[Login](#)

- 4) Login 클릭으로 인한 {SSO Server}/IDP/dispatch/login 로그인
- 5) 통합 인증 처리 이후 SAMPLE/TestSso.jsp 페이지로 이동됨.

TEST Redirection SSO

Hello [test1] :) You successfully logged in Rathon SSO service
[RedirectLogout](#)

Session Attributes

Key	Value
IDP_SESSION_PERSISTENCE	1567502412326
IDP_SESSION_ID	E29F0BD9D3E116FEDA0EF1919EF77E48FD353BB868898742E9684FB812EBFBEE
RathonSSO_UniqueID	44432333CCE18FF71C43DC94B26B9517`1567502181873
RathonSSO_USER_ID	test1

Request Parameters

Key	Value
-----	-------

10.2 SSO 로그아웃

- 1) SAMPLE/TestLogout.jsp?RelayState={로그아웃 후 이동할 Target URL}

Hello [test1] :) You successfully logged in Rathon SSO service
[RedirectLogout](#)

- 2) {SSO Server}/IDP/logout 호출 - Single Logout 처리
- 3) '1)'의 RelayState 로 지정된 Target URL 로 Redirect

11 삭제

- 1) 설정, 로그 등 보관이 필요한 자료는 백업을 진행한다.
- 2) 시스템을 정지한다.
- 2) 배포한 RathonSSO_SP_JDK.jar 및 의존성 라이브러리를 제거한다.
- 3) logback.xml, rathon_sso_sp.properties 설정 파일을 제거한다.

12 문제해결

12.1 SSO 서버와의 연결 문제

- 1) rathon_sso_sp.properties 설정 파일의 IDPM_DOMAIN_CONTEXT 설정에 주소 정보가 정확하게 입력되었는지 확인한다.

```
## SSO Server URL
IDPM_DOMAIN_CONTEXT=http://127.0.0.1:8080/IDP
```

12.2 등록되지 않는 시스템 문제

- 1) rathon_sso_sp.properties 설정 파일의 AGENT_PROVIDER_NAME 설정에 정보가 정확하게 입력되었는지 확인한다. 해당 정보는 SSO 관리자로부터 전달 받는다.
- 2) ID 가 정확한 경우 SSO 서버에 등록된 연동 시스템의 도메인 정보가 정확한지 확인한다.

```
## Agent ID
AGENT_PROVIDER_NAME=SAMPLE_SYSTEM
```


12.3 요청 데이터 변조 문제

1) rathon_sso_sp.properties 설정 파일의 AGENT_SIG_ALGORITHM, AGENT_SIG_ALG_KEY 설정에 정보가 정확하게 입력되었는지 확인한다. 해당 정보는 SSO 관리자로부터 전달 받는다.

2) 입력값이 정확한 경우 SSO 관리자에게 재확인 요청을 한다.

```
## HMAC-Signature-Algorithm
## HmacSHA256, HmacSHA384, HmacSHA512
AGENT_SIG_ALGORITHM=HmacSHA512
## HMAC-Signature-Key
AGENT_SIG_ALG_KEY=3a5fca31cceed0e6d61b022984ca64cec03b8b4df158d422c82d557dbeef415b
```

12.4 요청 무한루프 문제

- 1) SSO 인증을 받은 사용자가 연동 시스템에 접근시 SSO 서버 <-> 연동 시스템간 무한 루프에 빠지는 현상
- 2) 연동 시스템의 HTTPS 를 Proxy 방식으로 연동시 발생할 수 있음
- 3) 연동 시스템의 웹서버까지는 HTTPS 연결, 웹서버에서 WAS 까지의 연결은 HTTP 를 사용
- 4) 또는 동일한 도메인의 HTTPS, HTTP 프로토콜을 모두 사용

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite
ASP.NET_SessionId	zx4gu1xztmvqc12iwgo50ufn	sp.rathontech.com	/	Session	41	✓		
JSESSIONID	1763E385EC87FFD54DC9454F520A316D	sp.rathontech.com	/SP	Session	42		✓	

- 5) HTTPS 에서 생성된 쿠키가 Secure 설정이 생성되어서 발생하는 문제
- 6) 모든 통신을 HTTPS 만을 사용을 권장함
- 7) 또는, 연동 시스템의 HTTP 와 HTTPS 의 세션이 공유되도록 수정

12.5 응답 값 소실 문제

- 1) HTTPS 를 사용하는 연동시스템에서
- 2) SSO 서버에서 응답을 보낼 주소를 http://system.com 으로 알고 보냈을 때
- 3) 연동시스템은 HTTP 의 요청을 자동으로 HTTPS 로 변경시 302 Redirect 가 발생
- 4) 302 Redirect 시 POST Method 가 GET Method 로 변경되며 파라미터가 소실됨
- 5) 웹서버 등에서 307 Temporary Redirect 을 하도록 수정해 주면 해결됨

문 서 명	Rathon-SSO v4.3 Java 연동 매뉴얼
작 성 자	조영민
작 성 일	2021. 11. 01.

12.6 Single Logout 문제

- 1) SSO 가 연동된 여러 시스템 사용시 Single Logout 기능을 호출하였으나 특정 시스템이 로그아웃되지 않는 문제
- 2) rathon_sso_sp.properties 설정 파일의 SPM_LOGOUT_RES_URI 설정에 연동 시스템의 로그아웃 URL 정보가 정확하게 입력되었는지 확인한다.

```
## Agent Logout URL (SSO Server -> Agent)##  
## Servlet: {contextPath}/dispatch/slo (Relative or Absolute)##  
## Custom Page path (Relative or Absolute)##  
SPM_LOGOUT_RES_URI=/dispatch/slo##
```

- 3) HTTPS Proxy 연동으로 SSO 서버가 HTTP 로 주소를 인식하는 경우 Full Path 입력등으로 대처한다.
- 4) 주소가 정확하나 기본 제공된 페이지가 아닌 별도 경로를 지정한 경우 연동 시스템의 로그아웃 페이지의 기능이 오류가 발생하는지 체크한다.
- 5) SSO 서버 관리자에게 문의하여 SSO 서버가 대상 시스템의 도메인을 못찾거나 호출에 실패한 경우 네임서버에 도메인 등록 또는 방화벽 개방 절차를 수행한다.