Casablanca AAI Security/Vulnerability Report

This table represents the known exploitable and non-exploitable vulnerabilities in third party packages used in the project.

Repository	Group	Impact Analysis	Action
• aai /model-loader • aai/babel • aai /sparky-be • aai/data-router • aai/aai-resources • aai/aai-traversal • aai /event-client • aai /gizmo • aai /champ • aai /validation	com. fasterxml .jackson. core	False Positive. The exploit primarily is about enabling polymorphic type handling with the object mapper and writing class specifics into the JSON object. There are two ways of doing this: 1. ObjectMapper.enableDefaultTyping() 2. @JsonTypeInfo for marshalling / unmarshalling an object By default the ObjectMapper does not enableDefaultTyping, the code base is not using either approach, so the possibility of the exploit vector does not apply.	

• aai /event-client	com. fasterxml .jackson. core	DMaaP client dependency:
		
		`[INFO] +- org.onap.dmaap.messagerouter.dmaapclient:dmaapClient:jar:1.1.5: compile`
		
		`[INFO] | +- com.fasterxml.jackson.core:jackson-core:jar:2.8.11:compile`
		
		`[INFO] | +- com.fasterxml.jackson.core:jackson-databind:jar:2.8.11.1:compile`
		
		From Dmaap Security/Vulnerability - Beijing: The application is vulnerable by using this component, when default typing is enabled. Message Router do not use the default typing, so using the jackson-databind will not make message router vulnerable	

aai/champ	com. fasterxml .jackson. core	False Positive. The exploit primarily is about enabling polymorphic type handling with the object mapper and writing class specifics into the JSON object. There are two ways of doing this: 1. ObjectMapper.enableDefaultTyping() 2. @JsonTypeInfo for marshalling / unmarshalling an object By default the ObjectMapper does not enableDefaultTyping, the code base is not using either approach, so the possibility of the exploit vector does not apply.	
aai/aai-common	com. fasterxml .jackson. core	False Positive. The exploit primarily is about enabling polymorphic type handling with the object mapper and writing class specifics into the JSON object. There are two ways of doing this: 1. ObjectMapper.enableDefaultTyping() 2. @JsonTypeInfo for marshalling / unmarshalling an object By default the ObjectMapper does not enableDefaultTyping, the code base is not using either approach, so the possibility of the exploit vector does not apply.	

• aai/aai-resouces • aai/aai-traversal • aai /champ	org.codehaus.jackson	False Positive. The exploit primarily is about enabling polymorphic type handling with the object mapper and writing class specifics into the JSON object. There are two ways of doing this: 1. ObjectMapper.enableDefaultTyping() 2. @JsonTypeInfo for marshalling / unmarshalling an object By default the ObjectMapper does not enableDefaultTyping, the resources code bases are not using either approach, so the possibility of the exploit vector does not apply.	~~AAI-900~~ - Security: CVE-2017-7525 jackson-mapper-asl 1.9.2 CLOSED
• aai /champ	org.codehaus.jackson	False Positive. The exploit primarily is about enabling polymorphic type handling with the object mapper and writing class specifics into the JSON object. There are two ways of doing this: 1. ObjectMapper.enableDefaultTyping() 2. @JsonTypeInfo for marshalling / unmarshalling an object By default the ObjectMapper does not enableDefaultTyping, the resources code bases are not using either approach, so the possibility of the exploit vector does not apply.	
aai/aai-common	org.codehaus.jackson	False Positive. The exploit primarily is about enabling polymorphic type handling with the object mapper and writing class specifics into the JSON object. There are two ways of doing this: 1. ObjectMapper.enableDefaultTyping() 2. @JsonTypeInfo for marshalling / unmarshalling an object By default the ObjectMapper does not enableDefaultTyping, the code base is not using either approach, so the possibility of the exploit vector does not apply.	
aai/search-data-service	com.fasterxml.jackson.core	False Positive. The exploit primarily is about enabling polymorphic type handling with the object mapper and writing class specifics into the JSON object. There are two ways of doing this: 1. ObjectMapper.enableDefaultTyping() 2. @JsonTypeInfo for marshalling / unmarshalling an object By default the ObjectMapper does not enableDefaultTyping, the search service is not using either approach, so the possibility of the exploit vector does not apply.	

aai/esr-server	com.fasterxml.jackson.core	False Positive Explanation: This vulnerability issue only exists if com.fasterxml.jackson.databind.ObjectMapper.setDefaultTyping() is called before it is used for deserialization. esr-server doesn't invoke this method, esr-server use new Gson().fromJson(String json, Obj.class) and new Gson().toJson(obj) to deserialization and serialization. https://github.com/FasterXML/jackson-docs/wiki/JacksonPolymorphicDeserialization In esr-server, Gson is used to deserialization and serialization: https://gerrit.onap.org/r/gitweb?p=aai/esr-server.git;a=blob;f=esr-mgr/src/main/java/org/onap/aai/esr/wrapper/EmsManagerWrapper.java;h=588baad96c7942e83e0670784bbf423505c7b194;hb=HEAD https://gerrit.onap.org/r/gitweb?p=aai/esr-server.git;a=blob;f=esr-mgr/src/main/java/org/onap/aai/esr/wrapper/ThirdpartySdncWrapper.java;h=874205920c156f12df0bc591638a24e3f5575c76;hb=HEAD https://gerrit.onap.org/r/gitweb?p=aai/esr-server.git;a=blob;f=esr-mgr/src/main/java/org/onap/aai/esr/wrapper/VimManagerWrapper.java;h=fe44536cecb3f9ae9eaa3d99ff7b2d52511e2d52;hb=HEAD https://gerrit.onap.org/r/gitweb?p=aai/esr-server.git;a=blob;f=esr-mgr/src/main/java/org/onap/aai/esr/wrapper/VnfmManagerWrapper.java;h=8c7c5d39ceadff5e17f9c6d26d5540be49ada070;hb=HEAD https://gerrit.onap.org/r/gitweb?p=aai/esr-server.git;a=blob;f=esr-mgr/src/main/java/org/onap/aai/esr/util/ExtsysUtil.java;h=3bd01772356055e9711705b8518d55f1678b5179;hb=HEAD	
• aai/aai-resources • aai/aai-traversal • aai/aai-common	org.apache.activemq	Will update in Casablanca Maintenance Release. Issue is a false positive. This vulnerability is dependent on XalanXPathEvaluator.java using an insecure or absent document parser. AAI is not using this class.	~~AAI-1931~~ - Update to activemq-broker 5.15.8 CLOSED
aai/cacher	org.apache.activemq	Will update in Casablanca Maintenance Release. Issue is a false positive. This vulnerability is dependent on XalanXPathEvaluator.java using an insecure or absent document parser. AAI is not using this class.	~~AAI-1934~~ - [cacher] Update activemq-broker to 5.15.8 CLOSED ~~AAI-1935~~ - [cacher] cherry-pick update activemq-broker to 5.15.8 to Casablanca CLOSED
aai/cacher	org.apache.activemq	Will update in Casablanca Maintenance Release. Application is vulnerable to the vulnerability, users should secure the system so users cannot snoop network traffic between cacher and the other end of the queue; an old version of aai-common has the import, and cacher should move to the latest, 1.3.2 (see JIRA tickets)	~~AAI-1936~~ - [cacher] update aai-core to 1.3.4 CLOSED ~~AAI-1937~~ - [cacher] Update aai-core to 1.3.2 or 1.4.0-SNAPSHOT CLOSED
aai/champ	commons-httpclient	False positive. This is imported by hadoop which is used for hbase configs; in Beijing, AAI is configured with Janus on cassandra so it will not be accessing these classes. In Casablanca, Champ will serve as a multi-purpose data broker so we will look to upgrade the hadoop libraries to the most current versions.	
aai/aai-esr-gui	org.webjars.npm bootstrap	False positive. The data-target attribute in bootstrap.js interprets encoded HTML entities as standard HTML entities when data-target is based on user supplied input. data-target attribute is not used	Helpdesk ticket 54851
aai/aai-esr-gui	org.webjars.npm bootstrap	False positive. The `show()`function in the `tooltip.js` file allows HTML and scripts in the `data-container` tooltip attribute values in the DOM elements without proper sanitization. The show() function is not used	
ai/champ	org.apache.hadoop	False positive. The ONAP system only use Janus on Casandra, so the hadoop libraries are never touched	~~AAI-1887~~ - [champ] [security] Hadoop vulnerability CLOSED

• aai/resources • aai/traversal • aai/champ • aai/gizmo • aai/search-data-service • aai/data-router • aai/sparky-be • aai/graphadmin • aai/cacher	org.apache.tomcat.embed	AAI is not vulnerable because tomcat is not used in these repos, jetty is the application server. This is a child dependency of springframework, JIRA ticket in next column will address it for the Casablanca Maintenance Release by updating to the latest spring boot.	~~AAI-1888~~ - Security: Springboot 1.5.15 has new nexusIQ critical exceptions CLOSED
• aai/resources • aai/traversal • aai/champ • aai/gizmo • aai/search-data-service • aai/data-router • aai/sparky-be • aai/graphadmin • aai/cacher • aai/aai-common	org.springframework	Will update in maintenance release when upgrading to latest spring boot. False positive. AAI is not serving static resources through the ResourceHttpRequestHandler.	~~AAI-1888~~ - Security: Springboot 1.5.15 has new nexusIQ critical exceptions CLOSED
• aai/aai-common • aai/resources • aai/traversal • aai/graphadmin	com.google.guava	This dependency is a child dependency of Cassandra which is required for the graphdb; newer versions of Cassandra do not upgrade to a non-vulnerable version of this depedency. Guava is vulnerable to Denial of Service (DoS) when untrusted input is supplied to the `AtomicDoubleArray` and `CompoundOrdering` cla sses - AAI doesn't depend on guava to do this anywhere. Non-vulnerable versions of guava are not backward compatible with the version used by Cassandra	
aai/search-data-service	com.google.guava	A dependency of a child dependency, json-schema-validator. Even the latest version of json-schema-validator does not have the required fix version for the above components.	
aai/search-data-service	com.googlecode.libphonenumber	A dependency of a child dependency, json-schema-validator. Even the latest version of json-schema-validator does not have the required fix version for the above components. AAI is not vulnerable to this issue in the dependency, it does not use the component in the way described.	
aai/search-data-service	javax.mail	A dependency of a child dependency, json-schema-validator. Even the latest version of json-schema-validator does not have the required fix version for the above components. AAI is not vulnerable to this issue in the dependency, it does not use the component in the way described.	
aai/search-data-service	org.springframework.security	Inherited from spring boot, will be fixed in the Casablanca Maintenance Release. Search data service is not vulnerable to the exploit vectors because it does not perform the functions outlined in the report.	~~AAI-1895~~ - [search-data-service] Update springboot to 1.5.18 in search-data-service CLOSED

aai/data-router	com.att.aft	Update to 3.1.200-oss for Casablanca Maintenance Release. data-router does not the hazel cast component so we are not vulnerable in the meantime.	~~AAI-1938~~ - [data-router] Update dme2 to 3.1.200-oss CLOSED
aai/esr-server	com.smoketurner.dropwizard	False Positive. The exploit primarily is about enabling polymorphic type handling with the object mapper and writing class specifics into the JSON object. There are two ways of doing this: 1. ObjectMapper.enableDefaultTyping() 2. @JsonTypeInfo for marshalling / unmarshalling an object By default the ObjectMapper does not enableDefaultTyping, the code base is not using either approach, so the possibility of the exploit vector does not apply.	~~AAI-1970~~ - [esr-server] Remove zipkin-example CLOSED
aai/esr-server	com.smoketurner.dropwizard	It's an example brought by indirect dependency. Because the codes are not called by ESR, so it has no impact to ESR and it's downstream project. However, it should and will be deleted from ESR dependencies in the next release.	
aai/event-client	com.rabbitmq	False positive. Event client in ONAP only uses DMaaP so the rabbitmq dependencies are never used.	~~AAI-1905~~ - [event-client] Security - com.rabbitmq has vulnerabilities CLOSED
aai/esr-gui	org.apache.tomcat	ESR GUI is vulnerable. Implementors should secure the system to prevent exploits. We will replace tomcat in the Casablanca Maintenance Release with a version that is not vulnerable.	~~AAI-1967~~ - [esr-gui] update Apache tomcat CLOSED
aai/esr-gui	org.apache.tomcat	ESR GUI is vulnerable. Implementors should secure the system to prevent exploits. We will replace tomcat in the Casablanca Maintenance Release with a version that is not vulnerable.	~~AAI-1967~~ - [esr-gui] update Apache tomcat CLOSED
aai/esr-gui	org.apache.tomcat	ESR GUI is vulnerable. Implementors should secure the system to prevent exploits. We will replace tomcat in the Casablanca Maintenance Release with a version that is not vulnerable.	~~AAI-1967~~ - [esr-gui] update Apache tomcat CLOSED
aai/esr-gui	org.apache.tomcat	ESR GUI is vulnerable. Implementors should secure the system to prevent exploits. We will replace tomcat in the Casablanca Maintenance Release with a version that is not vulnerable.	~~AAI-1967~~ - [esr-gui] update Apache tomcat CLOSED
aai/esr-gui	org.apache.tomcat	ESR GUI is vulnerable. Implementors should secure the system to prevent exploits. We will replace tomcat in the Casablanca Maintenance Release with a version that is not vulnerable.	~~AAI-1967~~ - [esr-gui] update Apache tomcat CLOSED
aai/esr-gui	org.apache.tomcat	ESR GUI is vulnerable. Implementors should secure the system to prevent exploits. We will replace tomcat in the Casablanca Maintenance Release with a version that is not vulnerable.	~~AAI-1967~~ - [esr-gui] update Apache tomcat CLOSED
aai/esr-gui	jquery	ESR GUI is vulnerable. Implementors should secure the system to prevent exploits. We will replace jquery in the Casablanca Maintenance Release with a version that is not vulnerable.	~~AAI-1968~~ - [esr-gui] Update jquery CLOSED
aai/esr-gui	bootstrap	ESR GUI is vulnerable. Implementors should secure the system to prevent exploits. We will replace bootstrap in the Casablanca Maintenance Release with a version that is not vulnerable.	~~AAI-1969~~ - [esr-gui] Update bootstrap.js CLOSED