Features	References	*Mitre (Justin Richer)*	*Salesforce (Chuck Mortimore)*	*IBM - Session extensions for OAuth 2 (Todd Lainhart)*	*Huddle (Bob Gregory)*	*OpenESPI - Smart Grid (Donald F. Coffin)*	*openid_connect gem (built w/ rack-oauth2) (Nov Matake)*	*A2P3 (BC Government - Dick Hardt)*	*Roland Hedberg*	*Ping Identity (Brian Campbell)*
Client Types:										
Client Type: confidential	RFC 6749 §2.1	y	Yes	Yes		1 ✓	supported	yes	yes	Yes* - We've built the AS role and some facilitating components for the RS role. But we've done no client software. Answers herein should be considered thusly.
Client Type: public	RFC 6749 §2.1	y	Yes	Yes		1	supported	no	yes	Yes
Client Profiles:										
Client Profile: web application	RFC 6749 §2.1	y: client & server	Yes	Yes		1 ✓	supported	yes	yes	To the extent that the various options in 6749 are supported
Client Profile: user-agent-based application	RFC 6749 §2.1	y: server	Yes	No		1	supported	no	no	To the extent that the various options in 6749 are supported
Client Profile: native application	RFC 6749 §2.1	y: server	No	yes		1	supported	no	no	To the extent that the various options in 6749 are supported
Grant Types ("flows"):										
Grant Type: Authorization Code ("code flow")	RFC 6749 §§1.3.1, 4.1	y	Yes	Yes		1 ✓	supported	no	yes	Yes/configurable
For what use cases is this grant type used?		default	Server to Server integration with User present. OpenID Connect Flows	Classic webapp login	This forms the majority of our use cases - we use it for our centralised login server, and for granting access to third party applications. All third party clients current use this flow, and our website itself is being refitted to use this flow.	All End-User Authorization Request	for confidential clients			
Grant Type: Implicit ("implicit flow")	RFC 6749 §§1.3.2, 4.2	y	Yes	No			supported	no	yes	Yes/configurable
For what use cases is this grant type used?		discouraged, used in testing	Mobile and Desktop clients				for public clients			
Grant Type: Resource Owner Password Credentials	RFC 6749 §§1.1.3, 4.3	y	Yes	Yes		1	not supported	no	yes	Yes/configurable
For what use cases is this grant type used?		not used	Server to Server integration	Native client w/o client credentials	This is used in testing, and was originally used to support the website.					
Grant Type: Client Credentials	RFC 6749 §§1.3.4, 4.4	y	No	Yes		✓	not supported	no	yes	Yes/configurable
For what use cases is this grant type used?		server-server comms, client jwt assertions		Application identity for specialized operations		To allow Client to retrieve "Bulk" updates				
Grant Type: SAML2 Bearer Assertion	saml2-bearer §2.1	n	Yes	No			not supported	no	no	Yes/configurable - In addtion to the AS, we support client side here with WS-Trust STS
For what use cases is this grant type used?			Server to Server integration							
Grant Type: JWT Bearer Assertion	jwt-bearer §2.1	y	Yes	No		1	not supported	yes	no	Not yet
For what use cases is this grant type used?		refresh of id tokens	Server to Server integration		This is currently used for trusted applications to obtain elevated privileges					
Are any additional grant types defined or used?	RFC 6749 §8.3	n	We support an additional SAML assertion flow that directly reuses the SAML Web SSO Response Format for reuse of existing federations	Yes. Session.			not supported	no	no	yes, for access token introspection: see http://documentation.pingidentity.com/display/PF66/Grant+Type+Parameters#GrantTypeParameters-1079271

What additional grant types are defined, and what do they do?		none	We have one proprietary grant type we use for allowing clients to bypass IP restrictions on a salesforce tenant. We're deprecating it though, as we've found a more standards based approach.	Session Code Grant - creating/joining sessions response_type "jazz_session_code" required on authorization request grant_type "urn:jazz:params:oauth:grant-type:session_code" required on token request token_type "urn:jazz:params:oauth:token-type:session" required on token response	none	none	none		none	We have a grant type for "token introspection." See http://documentation.pingidentity.com/display/PF66/Grant+Type+Parameters#GrantTypeParameters-1079271
Authorization Request Parameters:										
client_id	RFC 6749 §2.2	y	Yes			1 ✓	supported	in JWE		
Is client_id structured or opaque?		opaque	opaque	opaque	Opaque	opaque	opaque	structured - hostname	opaque	opaque
What procedures or API(s) are used to register clients?		admin ui, admin api, dynreg	Developers register via a developer portal, or clients are automatically created in some provisioning processes	REST interface for dynamic client registration. Similar to what Justin is proposing.	Out-of-band registration is performed by administrators	Manual & Dynamic Registration	pre-registered and OpenID Connect Dynamic Client Registration	user interface and custom API	OIDC dynamic registration, manual registration	static UI or REST API
redirect_uri	RFC 6749 §3.1.2	y	Yes			1 ✓	supported	no		
Is registration of redirect_uri value(s) required?		n	Yes	Yes	Yes	Yes	yes		no	yes
Is the redirect_uri parameter always required to be supplied in an authorization request?		n	Yes	Yes	Currently, but we may drop it since we don't allow dynamic redirect uris	Yes	no		no	No, not if a single unambiguous redirect URI is registered for the client.
What URI value is used for native client applications?		varies, usually "myapp://" scheme	We support any URI, but restrict issuance of refresh tokens if it's a web URL	None.	Predominantly OOB, but we encourage the use of custom scheme handlers.		custom schema URI			any custom scheme
scope	RFC 6749 §3.3	y	Yes	No	Currently unsupported	✓	supported	scope is URIs in JWE	yes	
Is there a default scope value, and if so, what?		y, configurable (defaults to "openid profile address email phone")	Yes - API, and Identity			No	no		no	yes, it's configurable
Are additional requirements imposed on scope usage?		yes, scopes must be registered with AS and clients must be given access to scopes	No			They must meet scoping requirements of RS for the End-User	no		no	supported scopes are configurable
state	RFC 6749 §4.1.1	y	Yes			✓	supported	no	yes	
What is the state parameter used for?		?	CSRF protection and State maintenace for client simplicity	Used for CSRF thwarts	State parameters are used by third party clients, usually to store customer-specific information since we only allow a single redirect uri. The website uses state to return to the correct uri after login.	Client <--> AS Synchronization	against CSRF		maintaining state	to be echoed back to the client
Is state always required to be supplied in an authorization request?		client always sends it, server echoes it if present	No	No	Nope	Yes	no		no	no
Are additional requirements imposed on state usage?		n	No			No	no		no	no
Are extra parameters contained in the authorization request ignored?		y	Yes	Yes	Yes	Yes	yes		yes	yes
Are additional authorization request parameters defined or required?		n	Yes	No		No	yes (OpenID Connect params)		yes	There are application specific parameters that may be used to help select the appropriate means of end-user authentication and possibly UI language. Many deployments/configurations will not need them but in some configurations their use can reduce the amount of user interaction needed.

What additional authorization request parameters are defined, and what do they do?		none	Only parameters from OpenID Connect	none		none	OpenID Connect request parameters (nonce, etc.)		none	Proprietary application parameters that might assist, depending on configuration and deployment options, in choosing the way the end user is authenticated.
Authorization Response Parameters:										
code	RFC 6749 §4.1.2	y	Yes			✓	supported			
Is code structured or opaque?		opaque	opaque	opaque	Opaque	opaque	opaque		opaque	opaque
What lifetime is given to a code?		10mins (configurable)	not disclosed	2 minutes	Single-use, currently lasts forever, but we intend to impose an expiry.	maximum of 5 minutes	5 minutes		default = 10 minutes	configurable
Are additional requirements imposed on the code?		n	we'll vary some behaviors around IP restrictions and other constraints in our system	No		No	no		no	one time use
Are additional authorization response parameters defined or used?		n	Yes - an ID parameter and an Instance URL specific to Salesforce	No		No	yes (id_token)		no	no
What additional authorization response parameters are defined, and what do they do?		none	none	jazz_session_code	none	none	OpenID Connect parameters (id_token, etc.)		none	none
Authorization Error Responses:										
error	RFC 6749 §4.1.2.1	y	Yes			✓	supported	yes		
Are any additional error code values defined or used?		n (don't think so)	Yes - we have additional errors related to Administrative authorization failure	No	No	No	no	yes	no	no (though the OpenID Connect errors will be)
error_description	RFC 6749 §4.1.2.1	y	Yes			✓	supported	yes		
Is the error_description used, and if so, how?		yes, human-readable error description for specific errors	Refining error code	Yes. Descriptive text.	Error description is used to give a developer-readable description of the problem	Implementation Dependent	not sure	explain error	For human consumption	Yes, in many/most cases to give additional information as appropriate.
error_uri	RFC 6749 §4.1.2.1	y	Yes			✓	not supported	no		
Is the error_uri used, and if so, how?		n	Pointing to more info	No.	We use error uris to signal more specific error information, eg. https://login.huddle.net/docs/errors#RevokedAccessGrant	Implementation Dependent			no	no
Are additional authorization error response parameters defined or used?		not parameters, but some portions speak specific HTTP codes as well	no			No	no		no	no
What additional authorization error response parameters are defined, and what do they do?		none	none	none	none	None currently. May change as final design efforts are completed.	none		none	none
Token Request Parameters (some overlap with Authorization Request):										
redirect_uri	RFC 6749 §4.1.3	y	Yes			✓	supported	no		
Is the redirect_uri parameter always required to be supplied in a token request?		n, can be preregistered	yes	Yes.	Yes, but we may drop this requirement.	Yes	no		if included in the authzrequest	No, if it wasn't present in the corresponding original authorization request.
Are additional token request parameters defined or used?		n	no	No	yes	No	no		no	no
What additional token request parameters are defined, and what do they do?		none	none	jazz_subject	We DO have a signed-request means of client authentication in progress. That's likely to look something like this: POST /token client_id=foo&grant_type=auth_code&code=abc123&keyuri=-https://secure.com/keys/1.pem&alg=ES521&sig=abc123	none	none		none	token in http://documentation.pingidentity.com/display/PF66/Grant+Type+Parameters#GrantTypeParameters-1079271 is the access token for "introspection"
Token Response Parameters (some overlap with Authorization Response)										
access_token	RFC 6749 §§1.4, 4.1.2	y	Yes			✓	supported	yes	yes	
What lifetime is given to access tokens granted from a code?		configurable per client, defaults to 1h	sliding window dependant on tenant configuration	Configurable.	Default is 20 minutes, but configurable per-client.	Implementation Dependent	24 hours	until revoked	default = 60 minutes	configurable
Is access_token structured or opaque?		structured JWT	opaque	Opaque	JWT	opaque	opaque	opaque	opaque	configurable

Are additional requirements imposed on access_token?		signed by server	yes - all sorts of admin policies control our sessions			No	no		no	no
token_type ("Bearer", etc.)	RFC 6749 §§7.1, 4.1.4	y	No			✓	supported	bearer		
What token_type values are supported?		Bearer	Bearer	Session	We're still draft 12, so no token_types for us. When we refresh, we'll be JWT bearer only.	Bearer	Bearer		bearer	Bearer (+ urn:pingidentity.com:oauth2:validated_token as a special case for access token introspection)
expires_in	RFC 6749 §§5.1, 4.1.4	y	No			✓	supported	no		
Is expires_in used for original access tokens, and if so, with what values?		yes, same timeout as access token (defaults to 1h)		Configurable.	Yes, with the expiry time of the AT in seconds.	Implementation Dependent	yes (24 hours since issued)		default = 60 minutes from the time it was issued	Yes, if the access token has a fixed expiration time. Token expiration is configurable.
refresh_token	RFC 6749 §§1.5, 4.1.2, 6	y	Yes	Not yet		✓	not supported	no		
Is refresh_token structured or opaque?		structured JWT	opaque		Opaque.	opaque			opaque	opaque
What lifetime is given to a refresh token?		configurable per client, defaults to not expiring (I think?)	Depends on Admin policy		Refresh tokens are either per-access grant or single-use. Single use tokens cause problems because an unreliable network connection can cause users to lose their AT and RT.	Implementation Dependent			unlimited	configurable
Under what conditions is a refresh token issued?		client asks for offline_access scope	If asked for, allowed by admin policy, and if flow allows it to be issued to the redirect uri		Either never, or on every refresh for standard flows. We prefer the latter from a security point of view, but it requires that the connection is reliable. For elevated privileges, we require that the application resubmit its assertion, and do not issue an RT.	To replace an expired Access Token			always	based on client configuration data
Are extra parameters contained in the token request ignored?		y	no		Yes.	Yes			yes	yes
Are additional token response parameters defined or used?		n	no		No.	Yes; Resource URI is supplied with Access Token			no	no
What additional token response parameters are defined, and what do they do?		none	none	"jazz_subject":"user1" - user principal "jazz_groups" : [...] - JEE roles	none	"resource_uri" parameter has been added to JSON response. It defines the URI of the resource the client has been granted access to by the resource owner.	OpenID Connect parameters (id_token, etc.)		none	client_id comes back from http://documentation.pingidentity.com/display/PF66/Grant+Type+Parameters#GrantTypeParameters-1079271
Token Error Responses:								yes		
Are additional token error response parameters defined or used?		n	Yes			No	no	no	no	no
What additional token error response parameters are defined, and what do they do?		none	none	none	none	None currently. May change as final design efforts are completed.	none		none	none
Refresh Request Parameters (some overlap with Token Request):				No				NA		
refresh_token	RFC 6749 §6	y	yes			1 ✓				
scope	RFC 6749 §6	y	yes			✓				
Is down-scoping access tokens from refresh tokens supported?		y	no		No.	No			yes	Yes
Is a scope value required in a refresh request?		n (defaults to equal scope)	no		No.	Yes			no	No
Are extra parameters contained in the refresh request ignored?		y	yes		Yes.	Yes			yes	yes
Are additional refresh request parameters defined or used?		n	no		No.	No			no	no
What additional refresh request parameters are defined, and what do they do?		none	none	Refresh not yet implemented	none	none	none		none	none
Refresh Response Parameters (some overlap with Token Response):				No				NA		
access_token	RFC 6749 §§1.4, 4.1.2	y	yes			✓				
What lifetime is given to access tokens granted from a refresh token?		configurable (?) defaults to 10 min	sliding window dependant on tenant configuration		Refreshed tokens have the same configurable expiry time as the original AT.	Implementation Dependent			default = 60 minutes	configurable
Are additional requirements imposed on access tokens?		n	no			No			no	no
token_type ("Bearer", etc.)	RFC 6749 §§7.1, 4.1.4	y	no			✓				
What token_type values are supported?		Bearer	Bearer			Bearer			bearer	bearer
expires_in	RFC 6749 §§5.1, 4.1.4	y	no			✓				

Is expires_in used for refreshed access tokens, and if so, with what values?		configurable, same as "new access tokens" above			Yes, with the expiry time of the AT in seconds.	Implementation Dependent			default = 60 minutes from the time it was issued	same as 66
refresh_token	RFC 6749 §§1.5, 4.1.2, 6	y	yes			✓				
Under what conditions is a refresh token issued?		if client has "reuse refresh tokens" flag un-set			Refresh token is always returned in a refresh request.	To replace an expired Access Token			never	new refresh token value is issued (or not) based on configuration policy
What lifetime is given to a refresh token?		configurable, defaults to not expiring				Implementation Dependent				configurable
Are additional refresh response parameters defined or used?		n	no			No			no	no
What additional refresh response parameters are defined, and what do they do?		none	none	Refresh not yet implemented	none	none	none		none	none
Refresh Error Responses:								NA		
Are additional refresh error response parameters defined or used?		n	no			No			no	no
What additional refresh error response parameters are defined, and what do they do?		none	none	Refresh not yet implemented	none	None currently. May change as final design efforts are completed.	none		none	none
Client Authentication Methods:										
Client Authentication: Password via HTTP Basic	RFC 6749 §2.3.1	y	no	Yes	No	✓	supported		yes	yes
Client Authentication: Password via Request Body	RFC 6749 §2.3.1	y	yes	No	Yes		supported		yes	yes
Client Authentication: Bearer Token	RFC 6750	n	no	No	Ish... we are currently working out a signed-request format for client authentication which is somewhere between a JWT and a SWT. We use E384 signatures over a set of key-vaue pairs. This means the HTTP request looks just like a usual request, but we get PK authentication of the client.		not supported		yes	No
Are bearer tokens used for client authentication structured or opaque?									opaque	na
What do bearer tokens used for client authentication contain?					client_id, key_uri, alg, kid, signature, and then the OAuth2 params.					na
Client Authentication: SAML2 Bearer Assertion	saml2-bearer §2.2	n	no	No			not supported		yes	no
Client Authentication: JWT Bearer Assertion	jwt-bearer §2.2	y, if client has "jwks_uri" field set and token is signed	no	No			not supported		yes	no
Protocol Endpoints:										
Protocol Endpoint: Authorization Endpoint	RFC 6749 §3.1	y	yes			✓	supported	no		
Are any additional authorization endpoint parameters defined or used?		n		Session support	We have a single auth endpoint, but it may choose to authenticate users via saml, 2-fqactor pin, or username/pw	No	yes (OpenID Connect params)		no	Isn't this the same question as 39?
Protocol Endpoint: Redirection Endpoint	RFC 6749 §3.1.2	n	yes		Unsupported	✓	supported	no		
Are redirection endpoints containing query parameters supported?			no	Yes		No	yes		yes	yes
Are redirection endpoints containing query parameters exposed?			no	No			??		no	?
Are any additional redirection endpoint parameters defined or used?		n	no	No		No	no		no	?
What additional redirection endpoint parameters are defined, and what do they do?		none	We have one well known endpoint that some active clients do. It doesn't do anything on its own.	Clients register a sign-out endpoint (that may get called on session revocation); Clients register their application root, that can be queried by other SSO members to know if it's safe to pass a session token to another member in the group;	none	None currently. May change as final design efforts are completed.	none		none	na
Protocol Endpoint: Token Endpoint	RFC 6749 §3.2	y	yes			✓	supported			
Is token endpoint stateful or stateless?		stateless	statefull	stateless	Stateless	Stateless	??		stateful	
Can the code be used more than once?		n	no	No	No.	No	no		no	no
Are any additional token endpoint parameters defined or used?		n	no	No	Yes, for elevated privilege.	No	no		no	yes, for extension grants
Are additional protocol endpoints defined or used?		introspection, revocation, client registration, userinfo	identity URL (similar to openid connect)		Originally we had a separate endpoint for refresh but this is deprecated.	Yes -- /Register & /Revoke	no		no	In Connect, which is in development

What additional protocol endpoints are utilized, and what do they do?		none	none	. REST/CRUD interface for client registration (not keeping up with current draft at present); . Introspection endpoint per latest draft - response augmented for jazz_subject/groups per token response; . Revocation endpoint per latest drafts; . Endpoint discovery, using JRD, loosely based on Webfinger; . Signin endpoint, similar to resource owner user/password grant; for native clients, client id not required;	none	Two additional protocol endpoints have been designed. /Revoke is used to allow a client to revoke access, refresh or registration tokens. /Registration is used to support dynamic client registration and registration parameter updates	openid connect endpoints (discovery & client registration)		none	OpenID Connect endpoints
response_type Combinations Supported:								NA		
code	RFC 6749 §4.1.1	y	yes	yes		1 ✓	supported		yes	yes
token	RFC 6749 §4.2.1	y	yes	yes		1	supported		yes	yes
id_token	multiple-response-types §3	n	no	no			supported		yes	yes
code id_token	multiple-response-types §5	n	no	no			supported		yes	yes
code token id_token	multiple-response-types §5	n	no	no			supported		yes	yes
token id_token	multiple-response-types §5	n	no	no			supported		yes	yes
code token id_token	multiple-response-types §5	n	no	no			supported		yes	yes
none	multiple-response-types §4	n	no	no			not supported		yes	no
What additional response_type values do you define, and what do they do?		none	none	response_type "jazz_session_code" required on authorization request	none		OpenID Connect response types		OpenID Connect response types	none at this time
scope Values Supported:								NA		
openid	OpenID Connect Messages §2.4	y	no	no			supported		yes	In Connect, which is in development
profile	OpenID Connect Messages §2.4	y	yes although not conformant yet	no			supported		yes	In Connect, which is in development
email	OpenID Connect Messages §2.4	y	no	no			supported		yes	In Connect, which is in development
address	OpenID Connect Messages §2.4	y	no	no			supported		yes	In Connect, which is in development
phone	OpenID Connect Messages §2.4	y	no	no			supported		yes	In Connect, which is in development
offline_access	OpenID Connect Messages §2.4	y	no	no			not supported		yes	In Connect, which is in development
What scope values do you use, and what do they mean?		configured via admin ui or admin api	id, api, chatter, visualforce, web, refresh_token. They represent access to various APIs / interfaces (with the exception of refresh_token)	A "test" scope for testing - a "default" scope which is implied (i.e. "default" equivalent to "all")	None at present, but some anticipated for granting access to single URIs, rather than broad roles or permission-sets	Structured scopes. See attached OpenESPI Scope Definition Document.	OpenID Connect scope values		openid	configurable
TLS (https:) Versions								**ANY TLS**		
What TLS version(s) do deployed endpoints use?		?		1.1	1.2	TLS 1.2/1.0	TLS 1.0		whatever openssl supports	not sure off the top of my head
What TLS version(s) are supported for other's endpoints?		?				TLS 1.2/1.0	any		whatever openssl supports	not sure off the top of my head
Bearer Token Transmission Methods										
Is the Authorization Request Header Field method supported?	RFC 6750 §2.1	y	yes	yes	yes	Yes	yes	yes	yes	yes
Is the Form-Encoded Body Parameter method supported?	RFC 6750 §2.2	y	no	no	yes	No	yes	yes	yes	yes
Is the URI Query Parameter method supported?	RFC 6750 §2.3	y	sometimes	no	and furthermore yes, but only as a workaround when all else fails	No	yes	yes	yes	yes
WWW-Authenticate Response Header Field										
Is the "realm" parameter supported?	RFC 6750 §3	y	no	yes	Nope		yes	no	yes	yes, sort of
Are additional requirements imposed on realm usage?		n					no		no	
Is a "scope" value returned in the WWW-Authenticate response?	RFC 6750 §3	?	no	no			no	no	no	yes, for end user endpoint
Under what conditions is a "scope" value included in the response?										
WWW-Authenticate Error Responses										
Are the invalid_request, invalid_token, and insufficient_scope errors supported?	RFC 6750 §3.1	y		yes	Yes		yes	no	no	yes
Are any additional WWW-Authenticate error responses defined?		n			Nooo		no			no
Is the error_description WWW-Authenticate response parameter supported?	RFC 6750 §3	y		yes	No		yes	no	no	yes

Under what circumstances is an error_description value returned?		all errors					always			when it makes sense to give more info
Is the error_uri WWW-Authenticate response parameter supported?	RFC 6750 §3	n		no	Yes, under all circumstances		no	no	no	no
Under what circumstances is an error_uri value returned?										na
Are any additional means of communicating resource error information used?		HTTP codes		yes	No		no	yes, json response		no
What additional means of communicating resource errors are defined, and what do they do?		none	none	none	none	none	none		none	none
HTTP Authentication Schemes										
What additional HTTP Authentication Schemes do you define or use, and what do they do?				X-Jazz-Session						
Using SAML Assertions with OAuth 2.0								no		
Is using SAML Assertions as Authorization Grants supported?	saml2-bearer §2.1	n	yes	no	Nope		no		no	yes
Under what circumstances are SAML Assertions used as Authorization Grants?			Server to Server integration							deployment decision
Are any requirements added when using SAML Assertions as Authorization Grants?			no							deployment decision
Is using SAML Assertions for Client Authentication supported?	saml2-bearer §2.2		no				no		no	no
Under what circumstances are SAML Assertions used for Client Authentication?										
Are any requirements added when using SAML Assertions for Client Authentication?										
SAML Assertion Contents when used with OAuth 2.0				No				no		
What value(s) are used for the Audience element?	saml2-bearer §3		token endpoint							entityid or token endpoint
What value is used for the Subject element?	saml2-bearer §3		the principal							configurable for deployment
What format is used for the NameID element?			email							configurable for deployment
What value is used for the NameID element?			salesforce username							configurable for deployment
What expiration time is used for SAML Assertions?	saml2-bearer §3		assertion defined with a server side window of 5 minutes							configurable for deployment
Is a NotBefore element used in SAML Assertions?	saml2-bearer §3		yes							no
Is an IssueInstant element used in SAML Assertions?	saml2-bearer §3		yes							Yes (it's required per SAML schema)
Are any additional SubjectConfirmationData elements added to SAML Assertions?	saml2-bearer §3		yes - audience							Just what's required for bearer conf
What additional SubjectConfirmationData elements are added to SAML Assertions?										
Are any additional Conditions added to SAML Assertions?	saml2-bearer §3		no							no
What additional Conditions are added to SAML Assertions?										
Is replay of SAML Assertions prevented?	saml2-bearer §3		yes							no
What value is used for the ID element?			opaque							
Under what circumstances is an AuthnStatement included in SAML Assertions?	saml2-bearer §3		not required							yes
What is the contents of the AuthnStatement in issued SAML Assertions?										
Is adding AttributeStatement elements in SAML Assertions used with OAuth 2.0 supported?	saml2-bearer §3		yes with our proprietary web sso flow							yes
Under what circumstances are AttributeStatement elements added?			provisioning users							configurable for deployment
What additional AttributeStatement elements are added?										configurable
What algorithm(s) are used to sign SAML Assertions used with OAuth 2.0?	saml2-bearer §3		standard							RSA + SHA
What algorithm(s) are used to encrypt SAML Assertions used with OAuth 2.0?	saml2-bearer §3		not supported							not supported
Using JWTs with OAuth 2.0								yes		
Is using JWTs as Authorization Grants supported?	jwt-bearer §2.1	y	yes	no	Nope		no	yes	no	no
Under what circumstances are JWTs used as Authorization Grants?		refresh of id tokens	Server to Server integration					all		
Are requirements added when using JWTs as Authorization Grants?		must be a valid id token	no					no		
Is using JWTs for Client Authentication supported?	jwt-bearer §2.2	y	no				no	no	no	no
Under what circumstances are JWTs used for Client Authentication?		client needs higher-assurance auth								
Are requirements added when using JWTs for Client Authentication?		client must have jwks_uri registered, token must be signed								
JWTs Contents when used with OAuth 2.0										
What value is used for the "iss" (issuer) claim?	jwt-bearer §3	configured issuer URL of server	client id		https://login.huddle.net or a similar client identifier in the elevated privilege scenario.			app ID		
What value is used for the "sub" (subject) claim?	jwt-bearer §3	Spring Security principal name	user principal to act as					subject ID		
What value(s) are used for the "aud" (audience) claim?	jwt-bearer §3	client_id for issued client	tijeb ebdpoint					resource ID		
What expiration time is used for JWTs?	jwt-bearer §3	same as access tokens	client defined with a server side window of 5 minuts		20 mins			5 minutes		
Is a "nbf" (not before) claim used in JWTs?	jwt-bearer §3	n						no		
Is an "iat" (issued at) claim used in JWTs?	jwt-bearer §3	y						no		
Is replay of JWTs prevented?	jwt-bearer §3	n	yes					no		
What value is used for the "jti" claim?		randomly-generated UUID						not used		
Is adding additional claims in JWTs used with OAuth 2.0 supported?	jwt-bearer §3	y	no					yes		

Under what circumstances are additional claims added?		we don't do it in practice, but could			indicate scope
What additional claims are added?					
What algorithm(s) are used to sign JWTs used with OAuth 2.0?	jwt-bearer §3	RS256/RS512	standard	ES521	HS512
What algorithm(s) are used to encrypt JWTs used with OAuth 2.0?	jwt-bearer §3	RSA SSA 1_5	not supported	None currently, but we may adopt encryption for some of our shadowy government clients.	A256CBC+HS512

Specification Links:

http://tools.ietf.org/html/rfc6749	RFC6749 (OAuth 2.0 Core)
http://tools.ietf.org/html/rfc6750	RFC6750 (OAuth 2.0 Bearer)
http://tools.ietf.org/html/draft-ietf-oauth-saml2-bearer	saml2-bearer (OAuth SAML Profile)
http://tools.ietf.org/html/draft-ietf-oauth-jwt-bearer	jwt-bearer (OAuth JWT Profile)
http://openid.net/specs/oauth-v2-multiple-response-types-1_0.html	oauth-multiple-response-types
http://openid.net/specs/openid-connect-messages-1_0.html	OpenID Connect Messages