

Commit

✓ **fix: AdmissionPolicyGroup must not have context aware resources**

[Browse files](#)

Prior to this commit, it was possible to declare AdmissionPolicyGroup resources with policies that had context aware resources.

Context awareness allows a policy to query the Kubernetes API with ``list`` and ``get`` verbs. The queries are run using the ServiceAccount of the Policy Server instance that hosts the policy.

AdmissionPolicyGroup are namespaced CRDs which can be created by non-admin Kubernetes users. The ability to deploy AdmissionPolicyGroup with context aware resources exposes the cluster to potential leaks of data. For example, a non-admin user, might create an AdmissionPolicyGroup with one of its policies that has access to Kubernetes Secrets. The policy would then be able to ``get`` Secrets that are not available to the non-admin user (if the Policy Server is running with a ServiceAccount that has been explicitly configured to have ``get`` access to all the Secrets of the cluster).

This commit fixes the by removing the ``ContextAwareResources`` field from the ``AdmissionPolicyGroup``.

== What happens after this commit is deployed

The existing AdmissionPolicyGroup will be automatically updated and they will lose their ``.spec.policies[*].ContextAwareResources`` field (if set).

The kubewarden controller will also automatically reconcile the Policy Server configuration.

Signed-off-by: Flavio Castelli <fcastelli@suse.com>

 **main** (#984)
 **v1.21.0** v1.21.0-rc2

 **flavio** committed 2 weeks ago Verified 1 parent 04225f4 commit 51a88df

 Showing **13 changed files** with **522 additions** and **237 deletions**. Whitesp... Ignore whitesp... S... Uni...

 Filter changed files

▼  19  

api/policies/v1/admissionpolicygroup_types.go 

- api/policies/v1
 - admissionpolicygro... 
 - clusteradmissionpol... 
 - factories.go 
 - policy.go 
 - policy_types.go 
 - policygroup_validati... 
 - policygroup_validati... 
 - zz_generated.deepc... 
- config/crd/bases
 - policies.kubewarde... 
- docs/crds
 - CRD-docs-for-docs-r... 
 - CRD-docs-for-docs-r... 
- internal/controller
 - policyserver_control... 
 - policyserver_control... 

```

75     }
76
77     func (r *AdmissionPolicyGroup) IsContextAware()
78     bool {
79         -         for _, policy := range r.Spec.Policies {
80             -             if
81                 len(policy.ContextAwareResources) > 0 {
82                     -                 return true
83
84             -         }
85         }
86
87         +         // We return false here because
88         +         AdmissionPolicyGroup have no access to context
89         +         aware resources.
90
91         -         return false
92     }
93
94     - func (r *AdmissionPolicyGroup)
95     + func (r *AdmissionPolicyGroup)
96     + GetPolicyGroupMembers() PolicyGroupMembers {
97         -         return r.Spec.Policies
98     + func (r *AdmissionPolicyGroup)
99     + GetPolicyGroupMembersWithContext()
100     + PolicyGroupMembersWithContext {
101         +         policies :=
102         +         make(PolicyGroupMembersWithContext)
103         +         for name, policy := range r.Spec.Policies
104         +         {
105             +             policies[name] =
106             +             PolicyGroupMemberWithContext{
107                 +                 PolicyGroupMember: policy,
108                 +                 // AdmissionPolicyGroup
109                 +                 does not have access to context aware resources.
110                 +                 ContextAwareResources:
111                 +                 []ContextAwareResource{},
112             +             }
113         +         }
114         +         return policies
115     }
116
117     func (r *AdmissionPolicyGroup) GetSettings()
118     runtime.RawExtension {

```

 14 

api/policies/v1/clusteradmissionpolicygroup_types.go 

```

22     "k8s.io/apimachinery/pkg/runtime"
23 )
24
25 + type ClusterPolicyGroupSpec struct {
26 +     GroupSpec `json:""`

```

27	27	+	
28	28	+	// Policies is a list of policies that are part of the group that will
29	29	+	// be available to be called in the evaluation expression field.
30	30	+	// Each policy in the group should be a Kubewarden policy.
31	31	+	// +kubebuilder:validation:Required
32	32	+	<u>Policies</u> PolicyGroupMembersWithContext`json:"policies"`
33	33	+	}
34	34	+	
25	35		// ClusterAdmissionPolicyGroupSpec defines the desired state of ClusterAdmissionPolicyGroup.
26	36	type	ClusterAdmissionPolicyGroupSpec struct {
27	37	-	<u>PolicyGroupSpec</u> `json:""`
37	37	+	<u>ClusterPolicyGroupSpec</u> `json:""`
38	38		
29	39		// NamespaceSelector decides whether to run the webhook on an object based
30	40		// on whether the namespace for that object matches the selector. If the
148	158		return &r.Status
149	159		}
150	160		
151	161	-	func (r *ClusterAdmissionPolicyGroup) GetPolicyGroupMembers() <u>PolicyGroupMembers</u> {
161	161	+	func (r *ClusterAdmissionPolicyGroup) GetPolicyGroupMembersWithContext() <u>PolicyGroupMembersWithContext</u> {
152	162		return r.Spec.Policies
153	163		}
154	164		

44  api/policies/v1/factories.go 

314	314		},
315	315		Spec: AdmissionPolicyGroupSpec{
316	316		PolicyGroupSpec:
			PolicyGroupSpec{
317		-	PolicyServer:
			f.policyServer,
318		-	Policies:
			f.policyMembers,
319		-	Expression:
			f.expression,
320		-	Rules:
			f.rules,
321		-	MatchConditions:
			f.matchConds,
322		-	Mode:
			f.mode,

	317	+	GroupSpec{	GroupSpec:
			GroupSpec{	
	318	+	PolicyServer: f.policyServer,	
			PolicyServer: f.policyServer,	
	319	+	Expression: f.expression,	
			Expression: f.expression,	
	320	+	f.rules,	Rules:
			f.rules,	
	321	+	MatchConditions: f.matchConds,	
			MatchConditions: f.matchConds,	
	322	+	f.mode,	Mode:
			f.mode,	
	323	+	},	
			},	
	324	+	Policies:	
			f.policyMembers,	
323	325		},	
324	326		},	
325	327		}	
330	332		policyServer string	
331	333		rules	
			[]admissionregistrationv1.RuleWithOperations	
332	334		expression string	
333	-		policyMembers <u>PolicyGroupMembers</u>	
	335	+	policyMembers	
			<u>PolicyGroupMembersWithContext</u>	
334	336		matchConds	
			[]admissionregistrationv1.MatchCondition	
335	337		mode PolicyMode	
336	338		}	
353	355		},	
354	356		},	
355	357		expression: "pod_privileged() &&	
			user_group_psp()",	
356	-		policyMembers: <u>PolicyGroupMembers{</u>	
	358	+	policyMembers:	
			<u>PolicyGroupMembersWithContext{</u>	
357	359		"pod_privileged": {	
358	-		Module:	
			"registry://ghcr.io/kubewarden/tests/pod-	
			privileged:v0.2.5",	
	360	+	PolicyGroupMember:	
			PolicyGroupMember{	
	361	+	Module:	
			"registry://ghcr.io/kubewarden/tests/pod-	
			privileged:v0.2.5",	
	362	+	},	
359	363		},	
360	364		"user_group_psp": {	
361	-		Module:	
			"registry://ghcr.io/kubewarden/tests/user-group-	
			psp:v0.4.9",	

```

365 + PolicyGroupMember{
366 + Module:
    "registry://ghcr.io/kubewarden/tests/user-group-
    psp:v0.4.9",
367 + },
362 368 },
363 369 },
364 370 matchConds:
    []admissionregistrationv1.MatchCondition{
378 384         return f
379 385     }
380 386
381 - func (f *ClusterAdmissionPolicyGroupFactory)
    WithMembers(members PolicyGroupMembers)
    *ClusterAdmissionPolicyGroupFactory {
387 + func (f *ClusterAdmissionPolicyGroupFactory)
    WithMembers(members PolicyGroupMembersWithContext)
    *ClusterAdmissionPolicyGroupFactory {
382 388         f.policyMembers = members
383 389         return f
384 390     }
413 419     },
414 420 },
415 421 Spec:
    ClusterAdmissionPolicyGroupSpec{
416 - PolicyGroupSpec:
        PolicyGroupSpec{
417 - PolicyServer:
            f.policyServer,
418 - Policies:
            f.policyMembers,
419 - Expression:
            f.expression,
420 - Rules:
            f.rules,
421 - MatchConditions:
            f.matchConds,
422 - Mode:
            f.mode,
422 + ClusterPolicyGroupSpec:
        ClusterPolicyGroupSpec{
423 + GroupSpec:
            GroupSpec{
424 + PolicyServer: f.policyServer,
425 + Expression: f.expression,
426 + Rules:
            f.rules,
427 + MatchConditions: f.matchConds,

```

	428	+	Mode:
		f.mode,	
	429	+	},
	430	+	Policies:
		f.policyMembers,	
423	431		},
424	432		},
425	433	}	

2  api/policies/v1/policy.go 

151	151	// +kubebuilder:object:generate:=false
152	152	type PolicyGroup interface {
153	153	Policy
154	154	- <u>GetPolicyGroupMembers() PolicyGroupMembers</u>
	154	+ <u>GetPolicyGroupMembersWithContext()</u>
		<u>PolicyGroupMembersWithContext</u>
155	155	GetExpression() string
156	156	GetMessage() string
157	157	}

12  api/policies/v1/policy_types.go 

170	170	//
		+kubebuilder:pruning:PreserveUnknownFields
171	171	// x-kubernetes-embedded-resource: false
172	172	Settings runtime.RawExtension
		`json:"settings,omitempty"`
	173	+ }
	174	+
	175	+ type PolicyGroupMembersWithContext
		map[string]PolicyGroupMemberWithContext
	176	+
	177	+ type PolicyGroupMemberWithContext struct {
	178	+ PolicyGroupMember `json:""`
173	179	
174	180	// List of Kubernetes resources the policy
		is allowed to access at evaluation time.
175	181	// Access to these resources is done using
		the `ServiceAccount` of the PolicyServer
178	184	ContextAwareResources
		[]ContextAwareResource
		`json:"contextAwareResources,omitempty"`
179	185	}
180	186	
181	187	- type PolicyGroupSpec struct {
	187	+ type GroupSpec struct {
182	188	// PolicyServer identifies an existing
		PolicyServer resource.
183	189	// +kubebuilder:default:=default
184	190	// +optional
302	308	// returned in the warning field of the
		response.

```

303      // +kubebuilder:validation:Required
304      Message string `json:"message"`
311  + }
312  +
313  + type PolicyGroupSpec struct {
314  +     GroupSpec `json:""`
315
316      // Policies is a list of policies that are
317      // part of the group that will
318      // be available to be called in the
319      // evaluation expression field.

```



6



api/policies/v1/policygroup_validation.go


```

56      // validatePolicyGroupMembers validates that a
57      // policy group has at least one policy member.
58      func validatePolicyGroupMembers(policyGroup
59      PolicyGroup) field.ErrorList {
60          var allErrors field.ErrorList
61          if
62          len(policyGroup.GetPolicyGroupMembers()) == 0 {
63              if
64              len(policyGroup.GetPolicyGroupMembersWithContext()
65              ) == 0 {
66                  allErrors = append(allErrors,
67                  field.Required(field.NewPath("spec").Child("polici
68                  es"), "policy groups must have at least one policy
69                  member"))
70              }
71          }
72          for memberName := range
73          policyGroup.GetPolicyGroupMembers() {
74              for memberName := range
75              policyGroup.GetPolicyGroupMembersWithContext() {
76                  _, matchReservedSymbol :=
77                  celReservedSymbols[memberName]
78                  if len(memberName) == 0 ||
79                  matchReservedSymbol ||
80                  !idenRegex.MatchString(memberName) {
81                      allErrors =
82                      append(allErrors,
83                      field.Invalid(field.NewPath("spec").Child("policie
84                      s"), memberName, "policy group member name is
85                      invalid"))
86              }
87          }
88          // Create a CEL environment with custom
89          // functions for each policy member
90          var opts []cel.EnvOption
91          for policyName := range
92          policyGroup.GetPolicyGroupMembers() {
93              for policyName := range
94              policyGroup.GetPolicyGroupMembersWithContext() {
95                  fn := cel.Function(policyName,
96                  cel.Overload(policyName, []*cel.Type{},

```

		types.BoolType))
86	86	opts = append(opts, fn)
87	87	}

✓ ↕ 150 ■■■■■

api/policies/v1/policygroup_validation_test.go 

21	21	Name:
		"testing-cluster-policy-group",
22	22	},
23	23	Spec:
		ClusterAdmissionPolicyGroupSpec{
24	-	
		PolicyGroupSpec: PolicyGroupSpec{
25	-	
		Expression: "policy1() && policy2()",
26	-	
		Message: "This is a test policy",
27	-	
		Policies: PolicyGroupMembers{
	24	+
		ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{
	25	+
		GroupSpec: GroupSpec{
	26	+
		Expression: "policy1() && policy2()",
	27	+
		Message: "This is a test policy",
	28	+
	29	+
		Policies: PolicyGroupMembersWithContext{
28	30	
		"policy1": {
29	-	
		Module: "ghcr.io/kubewarden/tests/user-group- psp:v0.4.9",
	31	+
		PolicyGroupMember: PolicyGroupMember{
	32	+
		Module: "ghcr.io/kubewarden/tests/user-group- psp:v0.4.9",
	33	+
		},
30	34	
		},
31	35	
		"policy2": {
32	-	
		Module: "ghcr.io/kubewarden/tests/safe- labels:v1.0.0",
	36	+
		PolicyGroupMember: PolicyGroupMember{

	37	+	Module: "ghcr.io/kubewarden/tests/safe-labels:v1.0.0",
	38	+	},
33	39		},
34	40		},
35	41		},
44	50		Name:
			"testing-cluster-policy-group",
45	51		},
46	52		Spec:
			ClusterAdmissionPolicyGroupSpec{
47	-		
			PolicyGroupSpec: PolicyGroupSpec{
48	-		
			Expression: "",
49	-		
			Message: "This is a test policy",
50	-		
			Policies: PolicyGroupMembers{
53	+		
			ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{
54	+		
			GroupSpec: GroupSpec{
55	+		
			Expression: "",
56	+		
			Message: "This is a test policy",
57	+		},
58	+		
			Policies: PolicyGroupMembersWithContext{
51	59		
			"policy1": {
52	-		
			Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
60	+		
			PolicyGroupMember: PolicyGroupMember{
61	+		
			Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
62	+		
			},
53	63		
			},
54	64		
			"policy2": {
55	-		
			Module: "ghcr.io/kubewarden/tests/safe-labels:v1.0.0",

	65	+	PolicyGroupMember: PolicyGroupMember{
	66	+	Module: "ghcr.io/kubewarden/tests/safe-
			labels:v1.0.0",
	67	+	},
56	68		},
57	69		},
58	70		},
67	79		Name:
			"testing-cluster-policy-group",
68	80		},
69	81		Spec:
			ClusterAdmissionPolicyGroupSpec{
70	-		PolicyGroupSpec: PolicyGroupSpec{
71	-		Expression: "123",
72	-		Message: "This is a test policy",
73	-		Policies: PolicyGroupMembers{
	82	+	ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{
	83	+	GroupSpec: GroupSpec{
	84	+	Expression: "123",
	85	+	Message: "This is a test policy",
	86	+	},
	87	+	Policies: PolicyGroupMembersWithContext{
74	88		"policy1": {
75	-		Module: "ghcr.io/kubewarden/tests/user-group-
			psp:v0.4.9",
	89	+	PolicyGroupMember: PolicyGroupMember{
	90	+	Module: "ghcr.io/kubewarden/tests/user-group-
			psp:v0.4.9",
	91	+	},
76	92		},
77	93		"policy2": {

78	-	Module: "ghcr.io/kubewarden/tests/safe-labels:v1.0.0",
	94	+ PolicyGroupMember: PolicyGroupMember{
	95	+ Module: "ghcr.io/kubewarden/tests/safe-labels:v1.0.0",
	96	+ },
79	97	},
80	98	},
81	99	},
90	108	Name:
		"testing-cluster-policy-group",
91	109	},
92	110	Spec:
		ClusterAdmissionPolicyGroupSpec{
93	-	
		PolicyGroupSpec: PolicyGroupSpec{
94	-	
		Expression: "2 > 1",
95	-	
		Message: "This is a test policy",
96	-	
		Policies: PolicyGroupMembers{
	111	+ ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{
	112	+ GroupSpec: GroupSpec{
	113	+ Expression: "2 > 1",
	114	+ Message: "This is a test policy",
	115	+ },
	116	+ Policies: PolicyGroupMembersWithContext{
97	117	"policy1": {
98	-	
		Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
	118	+ PolicyGroupMember: PolicyGroupMember{
	119	+ Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
	120	+ },
99	121	},

100	122	"policy2": {	
101		-	
		Module: "ghcr.io/kubewarden/tests/safe-	
		labels:v1.0.0",	
	123	+	
		PolicyGroupMember: PolicyGroupMember{	
	124	+	
		Module: "ghcr.io/kubewarden/tests/safe-	
		labels:v1.0.0",	
	125	+	
		},	
102	126		
		},	
103	127		},
104	128		},
134	158		Name:
		"testing-cluster-policy-group",	
135	159		},
136	160		Spec:
		ClusterAdmissionPolicyGroupSpec{	
137		-	
		PolicyGroupSpec: PolicyGroupSpec{	
138		-	
		Expression: "policy1() && policy2()",	
139		-	
		Message: "This is a test policy",	
140		-	
		Policies: PolicyGroupMembers{	
	161	+	
		ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{	
	162	+	
		GroupSpec: GroupSpec{	
	163	+	
		Expression: "policy1() && policy2()",	
	164	+	
		Message: "This is a test policy",	
	165	+	},
	166	+	
		Policies: PolicyGroupMembersWithContext{	
141	167		
		"policy1": {	
142		-	
		Module: "ghcr.io/kubewarden/tests/user-group-	
		psp:v0.4.9",	
	168	+	
		PolicyGroupMember: PolicyGroupMember{	
	169	+	
		Module: "ghcr.io/kubewarden/tests/user-group-	
		psp:v0.4.9",	
	170	+	
		},	

143	171	},
144	172	"policy2": {
145	-	Module: "ghcr.io/kubewarden/tests/safe-labels:v1.0.0",
	173	+ PolicyGroupMember: PolicyGroupMember{
	174	+ Module: "ghcr.io/kubewarden/tests/safe-labels:v1.0.0",
	175	+ },
146	176	},
147	177	},
148	178	},
157	187	Name:
		"testing-cluster-policy-group",
158	188	},
159	189	Spec:
		ClusterAdmissionPolicyGroupSpec{
160	-	<u>PolicyGroupSpec: PolicyGroupSpec{</u>
161	-	Policies: map[string] <u>PolicyGroupMember{}</u> ,
	190	+ <u>ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{</u>
	191	+ Policies:
		map[string] <u>PolicyGroupMemberWithContext{}</u> ,
162	192	},
163	193	},
164	194	},
171	201	Name:
		"testing-cluster-policy-group",
172	202	},
173	203	Spec:
		ClusterAdmissionPolicyGroupSpec{
174	-	<u>PolicyGroupSpec: PolicyGroupSpec{</u>
175	-	Policies: <u>PolicyGroupMembers{</u>
	204	+ <u>ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{</u>
	205	+ Policies: <u>PolicyGroupMembersWithContext{</u>
176	206	": {
177	-	Module: "ghcr.io/kubewarden/tests/user-group-

```

    psp:v0.4.9",
207 +
    PolicyGroupMember: PolicyGroupMember{
208 +
    Module: "ghcr.io/kubewarden/tests/user-group-
    psp:v0.4.9",
209 +
    },
178 210
    },
179 211
    },
180 212
    },
189 221
    Name:
    "testing-cluster-policy-group",
190 222
    },
191 223
    Spec:
    ClusterAdmissionPolicyGroupSpec{
192 -
    PolicyGroupSpec: PolicyGroupSpec{
193 -
    Policies: PolicyGroupMembers{
224 +
    ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{
225 +
    Policies: PolicyGroupMembersWithContext{
194 226
    "in": {
195 -
    Module: "ghcr.io/kubewarden/tests/user-group-
    psp:v0.4.9",
227 +
    PolicyGroupMember: PolicyGroupMember{
228 +
    Module: "ghcr.io/kubewarden/tests/user-group-
    psp:v0.4.9",
229 +
    },
196 230
    },
197 231
    },
198 232
    },
207 241
    Name:
    "testing-cluster-policy-group",
208 242
    },
209 243
    Spec:
    ClusterAdmissionPolicyGroupSpec{
210 -
    PolicyGroupSpec: PolicyGroupSpec{
211 -
    Policies: PolicyGroupMembers{
244 +
    ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{

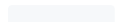
```

	245	+	Policies: <u>PolicyGroupMembersWithContext{</u>
212	246		"@policy1": {
213	-		Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
	247	+	PolicyGroupMember: PolicyGroupMember{
	248	+	Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
	249	+	},
214	250		},
215	251		},
216	252		},
225	261		Name:
			"testing-cluster-policy-group",
226	262		},
227	263		Spec:
			ClusterAdmissionPolicyGroupSpec{
228	-		<u>PolicyGroupSpec: PolicyGroupSpec{</u>
229	-		Policies: <u>PolicyGroupMembers{</u>
	264	+	<u>ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{</u>
	265	+	Policies: <u>PolicyGroupMembersWithContext{</u>
230	266		"!ol.ic?y1": {
231	-		Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
	267	+	PolicyGroupMember: PolicyGroupMember{
	268	+	Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
	269	+	},
232	270		},
233	271		},
234	272		},
243	281		Name:
			"testing-cluster-policy-group",
244	282		},
245	283		Spec:
			ClusterAdmissionPolicyGroupSpec{

246	-	<u>PolicyGroupSpec: PolicyGroupSpec{</u>	
247	-	Policies: <u>PolicyGroupMembers{</u>	
284	+	<u>ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{</u>	
285	+	Policies: <u>PolicyGroupMembersWithContext{</u>	
248	286	"_policy1": {	
249	-	Module: "ghcr.io/kubewarden/tests/user-group- psp:v0.4.9",	
287	+	PolicyGroupMember: PolicyGroupMember{	
288	+	Module: "ghcr.io/kubewarden/tests/user-group- psp:v0.4.9",	
289	+	},	
250	290	},	
251	291	"pol_icy2": {	
252	-	Module: "ghcr.io/kubewarden/tests/safe- labels:v1.0.0",	
292	+	PolicyGroupMember: PolicyGroupMember{	
293	+	Module: "ghcr.io/kubewarden/tests/safe- labels:v1.0.0",	
294	+	},	
253	295	},	
254	296		},
255	297		},
264	306		Name:
		"testing-cluster-policy-group",	
265	307		},
266	308		Spec:
		ClusterAdmissionPolicyGroupSpec{	
267	-	<u>PolicyGroupSpec: PolicyGroupSpec{</u>	
268	-	Policies: <u>PolicyGroupMembers{</u>	
309	+	<u>ClusterPolicyGroupSpec: ClusterPolicyGroupSpec{</u>	
310	+	Policies: <u>PolicyGroupMembersWithContext{</u>	

269	311	"po0licy1": {
270	-	Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
	312	+ PolicyGroupMember: PolicyGroupMember{
	313	+ Module: "ghcr.io/kubewarden/tests/user-group-psp:v0.4.9",
	314	+ },
271	315	},
272	316	"policy21": {
273	-	Module: "ghcr.io/kubewarden/tests/safe-labels:v1.0.0",
	317	+ PolicyGroupMember: PolicyGroupMember{
	318	+ Module: "ghcr.io/kubewarden/tests/safe-labels:v1.0.0",
	319	+ },
274	320	},
275	321	},
276	322	},

▼ 150  api/policies/v1/zz_generated.deepcopy.go 

 
 
  **Load diff**



▼  21 

config/crd/bases/policies.kubewarden.io_admissionpolicygroup... 

261	261	policies:
262	262	additionalProperties:
263	263	properties:
264	-	contextAwareResources:
265	-	description: -
266	-	List of Kubernetes
		resources the policy is allowed to access at
		evaluation time.

267		-	Access to these resources
		-	is done using the `ServiceAccount` of the
		-	PolicyServer
268		-	the policy is assigned to.
269		-	items:
270		-	description:
		-	ContextAwareResource identifies a Kubernetes
271		-	resource.
272		-	properties:
273		-	apiVersion:
274		-	description:
		-	apiVersion of the resource (v1 for core group,
275		-	groupName/groupVersions for other).
276		-	type: string
277		-	kind:
278		-	description: Singular
		-	PascalCase name of the resource
279		-	type: string
280		-	required:
281		-	- apiVersion
282		-	- kind
283		-	type: object
284		-	type: array
285	264		module:
286	265		description: -
287	266		Module is the location of
			the WASM module to be loaded. Can be a

✓  200  docs/crds/CRD-docs-for-docs-repo.adoc 

253	253		[cols="20a,50a,15a,15a", options="header"]
254	254		===
255	255		Field Description Default Validation
256		-	*` <u>PolicyGroupSpec</u> `* __xref:{anchor_prefix}-
			github-com-kubewarden-kubewarden-controller-api-
			policies-v1- <u>policygroupspec</u> [\$\$ <u>PolicyGroupSpec</u> \$_
	256	+	*` <u>ClusterPolicyGroupSpec</u> `* __xref:
			{anchor_prefix}-github-com-kubewarden-kubewarden-
			controller-api-policies-v1-
			<u>clusterpolicygroupspec</u> [\$\$ <u>ClusterPolicyGroupSpec</u> \$_]
			-
257	257		*`namespaceSelector`*
			__link:https://kubernetes.io/docs/reference/generat
			ed/kubernetes-api/v1.31/#labelselector-v1-
			meta[\$\$ <u>LabelSelector</u> \$_] NamespaceSelector
			decides whether to run the webhook on an object
			based +
258	258		on whether the namespace for that object matches
			the selector. If the +
259	259		object itself is a namespace, the matching is
			performed on +

```

393 393 |===
394 394
395 395
396 - [id="{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    contextawareresource"]
397 - ==== ContextAwareResource
    396 + [id="{anchor_prefix}-github-com-kubewarden-
        kubewarden-controller-api-policies-v1-
        clusterpolicygroupspec"]
    397 + ==== ClusterPolicyGroupSpec
    398 +
398 399
399 400
400 401
401 - ContextAwareResource identifies a Kubernetes
    resource.
402 402
403 403
404 404
405 405 .Appears In:
406 406 ****
407 - - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    clusteradmissionpolicyspec[$$ClusterAdmissionPolicy
    Spec$$]
408 - - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupmember[$$PolicyGroupMember$$]
    407 + - xref:{anchor_prefix}-github-com-kubewarden-
        kubewarden-controller-api-policies-v1-
        clusteradmissionpolicygroupspec[$$ClusterAdmissionP
        olicyGroupSpec$$]
409 408 ****
410 409
411 410 [cols="20a,50a,15a,15a", options="header"]
412 411 |===
413 412 | Field | Description | Default | Validation
414 - | *`apiVersion`* __string__ | apiVersion of the
    resource (v1 for core group,
    groupName/groupVersions for other). + | |
415 - | *`kind`* __string__ | Singular PascalCase name of
    the resource + | |
416 - |===
417 -
418 -
419 -
420 -
421 -
422 -
423 -
424 -

```

```

425 -
426 -
427 -
428 -
413 + | *`GroupSpec`* __xref:{anchor_prefix}-github-com-
      kubewarden-kubewarden-controller-api-policies-v1-
      groupspec[$$GroupSpec$$]__ | | |
414 + | *`policies`* __xref:{anchor_prefix}-github-com-
      kubewarden-kubewarden-controller-api-policies-v1-
      policygroupmemberswithcontext[$$PolicyGroupMembersW
      ithContext$$]__ | Policies is a list of policies
      that are part of the group that will +
415 + be available to be called in the evaluation
      expression field. +
416 + Each policy in the group should be a Kubewarden
      policy. + | | Required: {} +
429 417
      418 + |===
430 419
431 - [id="{anchor_prefix}-github-com-kubewarden-
      kubewarden-controller-api-policies-v1-
      policygroupmember"]
432 - ===== PolicyGroupMember
433 420
      421 + [id="{anchor_prefix}-github-com-kubewarden-
      kubewarden-controller-api-policies-v1-
      contextawareresource"]
      422 + ===== ContextAwareResource
434 423
435 424
436 425
      426 + ContextAwareResource identifies a Kubernetes
      resource.
437 427
438 428
439 429
440 430 .Appears In:
441 431 ****
442 - - xref:{anchor_prefix}-github-com-kubewarden-
      kubewarden-controller-api-policies-v1-
      policygroupmembers[$$PolicyGroupMembers$$]
432 + - xref:{anchor_prefix}-github-com-kubewarden-
      kubewarden-controller-api-policies-v1-
      clusteradmissionpolicyspec[$$ClusterAdmissionPolicy
      Spec$$]
433 + - xref:{anchor_prefix}-github-com-kubewarden-
      kubewarden-controller-api-policies-v1-
      policygroupmemberwithcontext[$$PolicyGroupMemberWit
      hContext$$]
443 434 ****
444 435
445 436 [cols="20a,50a,15a,15a", options="header"]

```

```

446 437 |===
447 438 | Field | Description | Default | Validation
448 - | *`module`* __string__ | Module is the location of
449 the WASM module to be loaded. Can be a +
450 - local file (file://), a remote file served by an
451 HTTP server +
452 - (http://, https://), or an artifact served by an
453 OCI-compatible +
454 - registry (registry://). +
455 - If prefix is missing, it will default to
456 registry:// and use that +
457 - internally. + | | Required: {} +
458 -
459 - | *`settings`*
460 __link:https://kubernetes.io/docs/reference/generat
461 ed/kubernetes-api/v1.31/#rawextension-runtime-
462 pkg[{$$RawExtension$$]__ | Settings is a free-form
463 object that contains the policy configuration +
464 - values. +
465 - x-kubernetes-embedded-resource: false + | |
466 - | *`contextAwareResources`* __xref:{anchor_prefix}-
467 github-com-kubewarden-kubewarden-controller-api-
468 policies-v1-
469 contextawareresource[{$$ContextAwareResource$$]
470 array__ | List of Kubernetes resources the policy
471 is allowed to access at evaluation time. +
472 - Access to these resources is done using the
473 `ServiceAccount` of the PolicyServer +
474 - the policy is assigned to. + | |
475 + | *`apiVersion`* __string__ | apiVersion of the
476 resource (v1 for core group,
477 groupName/groupVersions for other). + | |
478 + | *`kind`* __string__ | Singular PascalCase name of
479 the resource + | |
480 |===
481 441
482 442
483 443
484 - [id="{anchor_prefix}-github-com-kubewarden-
485 kubewarden-controller-api-policies-v1-
486 policygroupmembers"]
487 - ===== PolicyGroupMembers
488 -
489 - _Underlying type:_ __xref:{anchor_prefix}-github-
490 com-kubewarden-kubewarden-controller-api-policies-
491 v1-map-string-
492 policygroupmember[{$$map[string]PolicyGroupMember$$]
493 -
494 -
495 -
496 -
497 -
498 -

```

```

473 - .Appears In:
474 - ****
475 - - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupspec[$$PolicyGroupSpec$$]
476 - ****
477 -
478 -
479 -
480 - [id="{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupspec"]
481 - ==== PolicyGroupSpec
482 444 + [id="{anchor_prefix}-github-com-kubewarden-
483 447 kubewarden-controller-api-policies-v1-groupspec"]
484 448 + ==== GroupSpec
488 452
489 453 .Appears In:
490 454 ****
491 - - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    admissionpolicygroupspec[$$AdmissionPolicyGroupSpec
    $$]
492 - - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    clusteradmissionpolicygroupspec[$$ClusterAdmissionP
    olicyGroupSpec$$]
493 455 + - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    clusterpolicygroupspec[$$ClusterPolicyGroupSpec$$]
494 456 + - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupspec[$$PolicyGroupSpec$$]
495 457 ****
496 458
497 459 [cols="20a,50a,15a,15a", options="header"]
500 548 the policy group is rejected. The specific policy
    results will be +
501 549 returned in the warning field of the response. + |
    | Required: {} +
502 550
503 551 + |===
504 552 +
505 553 +
506 554 +
507 555 +
508 556 +
509 557 +
510 558 +

```

```

559 +
560 +
561 +
562 +
563 +
564 +
565 +
566 + [id="{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupmember"]
567 + ==== PolicyGroupMember
568 +
569 +
570 +
571 +
572 +
573 +
574 +
575 + .Appears In:
576 + ****
577 + - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupmemberwithcontext[$$PolicyGroupMemberWit
    hContext$$]
578 + - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupmembers[$$PolicyGroupMembers$$]
579 + ****
580 +
581 + [cols="20a,50a,15a,15a", options="header"]
582 + |===
583 + | Field | Description | Default | Validation
584 + | *`module`* __string__ | Module is the location of
    the WASM module to be loaded. Can be a +
585 + local file (file://), a remote file served by an
    HTTP server +
586 + (http://, https://), or an artifact served by an
    OCI-compatible +
587 + registry (registry://). +
588 + If prefix is missing, it will default to
    registry:// and use that +
589 + internally. + | | Required: {} +
590 +
591 + | *`settings`*
    __link:https://kubernetes.io/docs/reference/generat
    ed/kubernetes-api/v1.31/#rawextension-runtime-
    pkg[$$RawExtension$$]__ | Settings is a free-form
    object that contains the policy configuration +
592 + values. +
593 + x-kubernetes-embedded-resource: false + | |
594 + |===
595 +

```

```

596 +
597 + [id="{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupmemberwithcontext"]
598 + ==== PolicyGroupMemberWithContext
599 +
600 +
601 +
602 +
603 +
604 +
605 +
606 + .Appears In:
607 + ****
608 + - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupmemberswithcontext[$$PolicyGroupMembersw
    ithContext$$]
609 + ****
610 +
611 + [cols="20a,50a,15a,15a", options="header"]
612 + |===
613 + | Field | Description | Default | Validation
614 + | *`PolicyGroupMember`* __xref:{anchor_prefix}-
    github-com-kubewarden-kubewarden-controller-api-
    policies-v1-
    policygroupmember[$$PolicyGroupMember$$]__ | | |
615 + | *`contextAwareResources`* __xref:{anchor_prefix}-
    github-com-kubewarden-kubewarden-controller-api-
    policies-v1-
    contextawareresource[$$ContextAwareResource$$]
    array__ | List of Kubernetes resources the policy
    is allowed to access at evaluation time. +
616 + Access to these resources is done using the
    `ServiceAccount` of the PolicyServer +
617 + the policy is assigned to. + | |
618 + |===
619 +
620 +
621 + [id="{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupmembers"]
622 + ==== PolicyGroupMembers
623 +
624 + _Underlying type:_ __xref:{anchor_prefix}-github-
    com-kubewarden-kubewarden-controller-api-policies-
    v1-map-string-
    policygroupmember[$$map[string]PolicyGroupMember$$]
    -
625 +
626 +
627 +

```

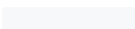
```

628 +
629 +
630 + .Appears In:
631 + ****
632 + - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupspec[$$PolicyGroupSpec$$]
633 + ****
634 +
635 +
636 +
637 + [id="{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupmemberswithcontext"]
638 + ==== PolicyGroupMembersWithContext
639 +
640 + _Underlying type:_ xref:{anchor_prefix}-github-
    com-kubewarden-kubewarden-controller-api-policies-
    v1-map-string-
    policygroupmemberwithcontext[$$map[string]PolicyGro
    upMemberWithContext$$]_
641 +
642 +
643 +
644 +
645 +
646 + .Appears In:
647 + ****
648 + - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    clusterpolicygroupspec[$$ClusterPolicyGroupSpec$$]
649 + ****
650 +
651 +
652 +
653 + [id="{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    policygroupspec"]
654 + ==== PolicyGroupSpec
655 +
656 +
657 +
658 +
659 +
660 +
661 +
662 + .Appears In:
663 + ****
664 + - xref:{anchor_prefix}-github-com-kubewarden-
    kubewarden-controller-api-policies-v1-
    admissionpolicygroupspec[$$AdmissionPolicyGroupSpec
    $$]

```

	665	+ ****
	666	+
	667	+ [cols="20a,50a,15a,15a", options="header"]
	668	+ ===
	669	+ Field Description Default Validation
	670	+ *`GroupSpec`* __xref:{anchor_prefix}-github-com-kubewarden-kubewarden-controller-api-policies-v1-groups spec[\$\$GroupSpec\$\$]__
587	671	*`policies`* __xref:{anchor_prefix}-github-com-kubewarden-kubewarden-controller-api-policies-v1-policygroupmembers[\$\$PolicyGroupMembers\$\$]__ Policies is a list of policies that are part of the group that will
588	672	be available to be called in the evaluation expression field. +
589	673	Each policy in the group should be a Kubewarden policy. + Required: {} +
607	691	
608	692	.Appears In:
609	693	****
610		- - xref:{anchor_prefix}-github-com-kubewarden-kubewarden-controller-api-policies-v1-policygroupspec[\$\$PolicyGroupSpec\$\$]
	694	+ - xref:{anchor_prefix}-github-com-kubewarden-kubewarden-controller-api-policies-v1-groups spec[\$\$GroupSpec\$\$]
611	695	- xref:{anchor_prefix}-github-com-kubewarden-kubewarden-controller-api-policies-v1-policyspec[\$\$PolicySpec\$\$]
612	696	****
613	697	

▼ 95  docs/crds/CRD-docs-for-docs-repo.md 



[Load diff](#)

Large diffs are not rendered by default.

▼  26 

internal/controller/policyserver_controller_configmap.go 

22	22	
23	23	const dataType string = "Data" // only data type is supported
24	24	
25		- type <u>policyGroupMember</u> struct {
	25	+ type <u>policyGroupMemberWithContext</u> struct {
26	26	Module string
		`json:"module"`

```

27         Settings runtime.RawExtension
27         `json:"settings,omitempty"`
28         ContextAwareResources
28         []policiesv1.ContextAwareResource
28         `json:"contextAwareResources,omitempty"`
36         ContextAwareResources
36         []policiesv1.ContextAwareResource
36         `json:"contextAwareResources,omitempty"`
37         Settings runtime.RawExtension
37         `json:"settings,omitempty"`
38         // The following fields are used by policy
38         groups only.
39         - Policies map[string]policyGroupMember
39         `json:"policies,omitempty"`
40         - Expression string
40         `json:"expression,omitempty"`
41         - Message string
41         `json:"message,omitempty"`
39         + Policies
39         map[string]policyGroupMemberWithContext
39         `json:"policies,omitempty"`
40         + Expression string
40         `json:"expression,omitempty"`
41         + Message string
41         `json:"message,omitempty"`
42     }
43
44     // The following MarshalJSON and UnmarshalJSON
44     methods are used to serialize
64     func (p policyServerConfigEntry) MarshalJSON()
64     ([]byte, error) {
65         if len(p.Policies) > 0 {
66             return json.Marshal(struct {
67                 - NamespacedName
67                 types.NamespacedName
67                 `json:"namespacedName"`
68                 - PolicyMode string
68                 `json:"policyMode"`
69                 - Policies
69                 map[string]policyGroupMember `json:"policies"`
70                 - Expression string
70                 `json:"expression"`
71                 - Message string
71                 `json:"message"`
67             + NamespacedName
67             types.NamespacedName
67             `json:"namespacedName"`
68             + PolicyMode string
68             `json:"policyMode"`
69             + Policies
69             map[string]policyGroupMemberWithContext
69             `json:"policies"`

```

	70	+	<code>`json:"expression"``</code>	Expression	string
	71	+	<code>`json:"message"``</code>	Message	string
72	72		<code>}{</code>		
73	73		<code>NamespacedName:</code>		
			<code>p.NamespacedName,</code>		
74	74		<code>PolicyMode:</code>		
			<code>p.PolicyMode,</code>		
174	174		<code>return</code>		
			<code>unstructuredObj.GetResourceVersion(), nil</code>		
175	175		<code>}</code>		
176	176				
177		-	<code>func buildPolicyGroupMembers(policies</code>		
			<code>policiesv1.PolicyGroupMembers)</code>		
			<code>map[string]policyGroupMember {</code>		
178		-	<code>policyGroupMembers :=</code>		
			<code>map[string]policyGroupMember{}</code>		
177		+	<code>func buildPolicyGroupMembersWithContext(policies</code>		
			<code>policiesv1.PolicyGroupMembersWithContext)</code>		
			<code>map[string]policyGroupMemberWithContext {</code>		
178		+	<code>policyGroupMembers :=</code>		
			<code>map[string]policyGroupMemberWithContext{}</code>		
179	179		<code>for name, policy := range policies {</code>		
180		-	<code>policyGroupMembers[name] =</code>		
			<code>policyGroupMember{</code>		
180		+	<code>policyGroupMembers[name] =</code>		
			<code>policyGroupMemberWithContext{</code>		
181	181		<code>Module:</code>		
			<code>policy.Module,</code>		
182	182		<code>Settings:</code>		
			<code>policy.Settings,</code>		
183	183		<code>ContextAwareResources:</code>		
			<code>policy.ContextAwareResources,</code>		
202	202		<code>}</code>		
203	203				
204	204		<code>if policyGroup, ok :=</code>		
			<code>admissionPolicy.(policiesv1.PolicyGroup); ok {</code>		
205		-	<code>configEntry.Policies =</code>		
			<code>buildPolicyGroupMembers(policyGroup.GetPolicyGroup</code>		
			<code>Members())</code>		
205		+	<code>configEntry.Policies =</code>		
			<code>buildPolicyGroupMembersWithContext(policyGroup.Get</code>		
			<code>PolicyGroupMembersWithContext())</code>		
206	206		<code>configEntry.Expression =</code>		
			<code>policyGroup.GetExpression()</code>		
207	207		<code>configEntry.Message =</code>		
			<code>policyGroup.GetMessage()</code>		
208	208		<code>}</code>		

276	276	
277	277	clusterPolicyGroup :=
		policiesv1.NewClusterAdmissionPolicyGroupFactory()
		.
278	278	WithPolicyServer(policyServerName).
279	-	WithMembers(policiesv1. <u>PolicyGroupMembers</u> {
	279	+ WithMembers(policiesv1. <u>PolicyGroupMembersWithConte</u>
		<u>xt</u> {
280	280	"pod_privileged": {
281	-	Module: "registry://ghcr.io/kubewarden/tests/pod-
		privileged:v0.2.5",
	281	+ PolicyGroupMember: policiesv1.PolicyGroupMember{
	282	+ Module: "registry://ghcr.io/kubewarden/tests/pod-
		privileged:v0.2.5",
	283	+ },
282	284	ContextAwareResources:
		[]policiesv1.ContextAwareResource{
283	285	{
284	286	APIVersion: "v1",
287	289	},
288	290	},
289	291	"user_group_psp": {
290	-	Module: "registry://ghcr.io/kubewarden/tests/user-
		group-ppsp:v0.4.9",
	292	+ PolicyGroupMember: policiesv1.PolicyGroupMember{
	293	+ Module: "registry://ghcr.io/kubewarden/tests/user-
		group-ppsp:v0.4.9",
	294	+ },
291	295	ContextAwareResources:
		[]policiesv1.ContextAwareResource{
292	296	{
293	297	APIVersion: "v1",
332	336	AllowedToMutate:
		admissionPolicyGroup.IsMutating(),

333	337	Settings:
		admissionPolicyGroup. <u>GetSettings()</u> ,
334	338	
		ContextAwareResources:
		admissionPolicyGroup. <u>GetContextAwareResources()</u> ,
335	-	Policies:
		<u>buildPolicyGroupMembers(admissionPolicyGroup.<u>GetPolicyGroupMembers()</u>),</u>
	339	+
		Policies:
		<u>buildPolicyGroupMembersWithContext(admissionPolicyGroup.<u>GetPolicyGroupMembersWithContext()</u>),</u>
336	340	Expression:
		admissionPolicyGroup. <u>GetExpression()</u> ,
337	341	Message:
		admissionPolicyGroup. <u>GetMessage()</u> ,
338	342	}
346	350	Settings:
		clusterPolicyGroup. <u>GetSettings()</u> ,
347	351	
		ContextAwareResources:
		clusterPolicyGroup. <u>GetContextAwareResources()</u> ,
348	352	PolicyMode:
		string(clusterPolicyGroup. <u>GetPolicyMode()</u>),
349	-	Policies:
		<u>buildPolicyGroupMembers(clusterPolicyGroup.<u>GetPolicyGroupMembers()</u>),</u>
	353	+
		Policies:
		<u>buildPolicyGroupMembersWithContext(clusterPolicyGroup.<u>GetPolicyGroupMembersWithContext()</u>),</u>
350	354	Expression:
		clusterPolicyGroup. <u>GetExpression()</u> ,
351	355	Message:
		clusterPolicyGroup. <u>GetMessage()</u> ,
352	356	}
406	410	}),
407	411	
		"policies": MatchKeys(IgnoreExtras, Keys{
408	412	
		"pod_privileged": MatchKeys(IgnoreExtras, Keys{
409	-	
		"module":
		Equal(admissionPolicyGroup. <u>GetPolicyGroupMembers()</u>
		["pod_privileged"].Module),
	413	+
		"module":
		Equal(admissionPolicyGroup. <u>GetPolicyGroupMembersWith</u>
		<u>WithContext()</u> ["pod_privileged"].Module),
410	414	}),
411	415	}),

412	416	"policyMode":
		Equal(string(admissionPolicyGroup.GetPolicyMode())
		),
420	424	}),
421	425	"policies": MatchKeys(IgnoreExtras, Keys{
422	426	"pod_privileged": MatchAllKeys(Keys{
423	-	"module":
		Equal(clusterPolicyGroup.GetPolicyGroupMembers()
		["pod_privileged"].Module),
	427	+ "module":
		Equal(clusterPolicyGroup.GetPolicyGroupMembersWith
		Context()["pod_privileged"].Module),
424	428	"settings": Ignore(),
425	429	"contextAwareResources":
		And(ContainElement(MatchAllKeys(Keys{
426	430	"apiVersion": Equal("v1"),
427	431	"kind": Equal("Pod"),
428	432	))), HaveLen(1)),
429	433	}),
430	434	"user_group_psp": MatchAllKeys(Keys{
431	-	"module":
		Equal(clusterPolicyGroup.GetPolicyGroupMembers()
		["user_group_psp"].Module),
	435	+ "module":
		Equal(clusterPolicyGroup.GetPolicyGroupMembersWith
		Context()["user_group_psp"].Module),
432	436	"settings": Ignore(),
433	437	"contextAwareResources":
		And(ContainElement(MatchAllKeys(Keys{
434	438	"apiVersion": Equal("v1"),

0 comments on commit 51a88df

Please [sign in](#) to comment.