

网管产品 - 错误 #4833

网管版本2.4.0pre1T1，界面参数rank、port、ant按照实际规则配置下发，无法配置

2026-02-03 09:55 - 杨凯

状态:	转测试	开始日期:	2026-02-03
优先级:	一般	计划完成日期:	
指派给:	马 党旗	% 完成:	0%
类别:		预期时间:	0.00 小时
目标版本:		耗时:	0.00 小时

## 历史记录

---

#1 - 2026-02-03 11:17 - 马党旗

- 状态从 新建 变更为 进行中

#2 - 2026-02-03 11:20 - 马党旗

- 文件 20260203-111911.jpg 已添加

- 状态从进行中变更为审视

已处理。原因有2：

1. 需要将原xpath值类型断言错误，
  2. 变量名复用问题

```
service/checkcfg.go (1a6dbd27)
339     param := map[string]any{}
340     if strings.Contains(name, ".") {
341         if v, ok := itemMap[name]; ok {
342             if v == nil {
343                 vstr, vok := v.(string)
344                 if vok {
345                     return nil, fmt.Errorf(`"\%s": "%s" => %s value [%v] is not a string `, name, rule, name, v)
346                 }
347                 iv, err := strconv.Atoi(vstr)
348                 if err != nil {
349                     return nil, err
350                 } else {
351                     param[name] = iv
352                 }
353             } else {
354                 param[name] = v
355             }
356         }
357     } else {
358         needInt := false
359         if hasOperations(rule) {
360             needInt = true
361         }
362         for k := range strings.SplitSeq(name, ".") {
363             if v, ok := itemMap[k]; ok {
364                 if needInt {
365                     if v == nil {
366                         return nil, fmt.Errorf(`"\%s": "%s" => %s value [%v] is not a string `, name, rule, k, v)
367                     }
368                     iv, err := strconv.Atoi(vstr)
369                     if err != nil {
370                         return nil, err
371                     } else {
372                         param[k] = iv
373                     }
374                 } else {
375                     param[k] = v
376                 }
377             }
378         }
379     }
380     if len(param) > 0 {
381         res, err := judgmentRes(rule, param)
382         if err != nil {
383             return nil, err
384         }
385         rtmp = append(rtmp, res)
386     }
387 }
388 if len(rtmp) > 0 {
389     res = append(res, rtmp)
390 }
391 }
392 listRes := []bool{}
393 for _, results := range res {
394     res := true
395     for _, r := range results {
396         if r == nil {
397             res = false
398             break
399         }
400     }
401 }
402 listRes = append(listRes, res)
403

service/checkcfg.go (Working Tree)
304     }
305 }
306 func parseRules(ruleStr []map[string]string, itemMap map[string]any) ([]bool, error) {
307     res := []bool{}
308     rtmp := []bool{}
309     for _, rules := range ruleStr {
310         rtmp = []bool{}
311         for name, rule := range rules {
312             param := map[string]any{}
313             if strings.Contains(name, ".") {
314                 if v, ok := itemMap[name]; ok {
315                     if hasOperations(rule) {
316                         if v == nil {
317                             iv, err := strconv.Atoi(fmt.Sprintf(v))
318                             if err != nil {
319                                 return nil, err
320                             } else {
321                                 param[name] = iv
322                             }
323                         } else {
324                             param[name] = fmt.Sprintf(v)
325                         }
326                     } else {
327                         needInt := false
328                         if hasOperations(rule) {
329                             needInt = true
330                         }
331                         for k := range strings.SplitSeq(name, ".") {
332                             if v, ok := itemMap[k]; ok {
333                                 if needInt {
334                                     iv, err := strconv.Atoi(fmt.Sprintf(v))
335                                     if err != nil {
336                                         return nil, err
337                                     } else {
338                                         param[k] = iv
339                                     }
340                                 } else {
341                                     param[k] = fmt.Sprintf(v)
342                                 }
343                             }
344                         }
345                         if len(param) > 0 {
346                             res, err := judgmentRes(rule, param)
347                             if err != nil {
348                                 return nil, err
349                             }
350                             rtmp = append(rtmp, res)
351                         }
352                     }
353                 }
354             }
355         }
356         if len(rtmp) > 0 {
357             res = append(res, rtmp)
358         }
359     }
360     if len(res) > 0 {
361         for _, results := range res {
362             rtmp := true
363             for _, r := range results {
364                 if r == nil {
365                     rtmp = false
366                 }
367             }
368             if rtmp {
369                 listRes = append(listRes, rtmp)
370             }
371         }
372     }
373 }
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
```

#3 - 2026-02-03 11:20 - 马党旗

- 状态从 审视 变更为 转测试

## 文件

---

20260203-111911.jpg

464 KB

2026-02-03

马党旗