

```

1  <?xml version='1.0' encoding='UTF-8'?>
2  <hotspot_log version='160 1' process='3805' time_ms='1578109306909'>
3  <vm_version>
4  <name>
5  OpenJDK 64-Bit Server VM
6  </name>
7  <release>
8  25.71-b00-debug
9  </release>
10 <info>
11 OpenJDK 64-Bit Server VM (25.71-b00-debug) for linux-amd64 JRE (1.8.0-internal-zbhuang_2019_11_03_07_31-b00), built on Jan  2 2020 22:16:34
12 by "zbhuang" with gcc 5.4.0 20160609
13 </info>
14 </vm_version>
15 <vm_arguments>
16 <args>
17 -XX:+UnlockDiagnosticVMOptions -XX:+PrintInterpreter -XX:+LogCompilation -XX:LogFile=Hello.PrintInterpreter.txt
18 </args>
19 <command>
20 Hello
21 </command>
22 <launcher>
23 SUN_STANDARD
24 </launcher>
25 <properties>
26 java.vm.specification.name=Java Virtual Machine Specification
27 java.vm.version=25.71-b00-debug
28 java.vm.name=OpenJDK 64-Bit Server VM
29 java.vm.info=mixed mode, sharing
30 java.ext.dirs=/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk/lib/ext:/usr/java/packages/lib/ext
31 java.endorsed.dirs=/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk/lib/endorsed
32 sun.boot.library.path=/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk/lib/amd64
33 java.library.path=/usr/java/packages/lib/amd64:/usr/lib64:/lib64:/lib:/usr/lib
34 java.home=/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk
35 java.class.path=./usr/local/java/jdk1.8.0_181/lib:/usr/local/java/jdk1.8.0_181/jre/lib:
36 sun.boot.class.path=/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk/lib/resources.jar:/home/zbhuang/OpenJDK/jdk8u/
37 build/linux-x86_64-normal-server-slowdebug/jdk/lib/rt.jar:/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk/lib/sunr
38 sasign.jar:/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk/lib/jsse.jar:/home/zbhuang/OpenJDK/jdk8u/build/linux-x8
39 6_64-normal-server-slowdebug/jdk/lib/jce.jar:/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk/lib/charsets.jar:/hom
40 e/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-server-slowdebug/jdk/lib/jfr.jar:/home/zbhuang/OpenJDK/jdk8u/build/linux-x86_64-normal-serv
41 er-slowdebug/jdk/classes
42 java.vm.specification.vendor=Oracle Corporation
43 java.vm.specification.version=1.8
44 java.vm.vendor=Oracle Corporation
45 sun.java.command=Hello
46 sun.java.launcher=SUN_STANDARD
47 </properties>
48 </vm_arguments>
49 <tty>
50 <writer thread='139642484700928' />
51 -----
52 Interpreter
53 -----
54 code size      =    320K bytes
55 total space    =   1023K bytes
56 wasted space   =    703K bytes
57 -----
58 # of codelets  =    266
59 avg codelet size =   1234 bytes
60 -----
61 method entry point (kind = zerolocals) [0x00007f00f5023be0, 0x00007f00f5024a00] 3616 bytes
62 -----
63 0x00007f00f5023be0: mov     0x10(%rbx),%rdx
64 0x00007f00f5023be4: movzwl 0x2c(%rdx),%ecx
65 0x00007f00f5023be8: movzwl 0x2a(%rdx),%edx
66 0x00007f00f5023bec: sub     %ecx,%edx
67 0x00007f00f5023bee: cmp     $0x1f6,%edx
68 0x00007f00f5023bf4: jbe     0x00007f00f5023d3d
69 0x00007f00f5023bfa: mov     %rdx,%rax
70 0x00007f00f5023bfd: shl     $0x3,%rax
71 0x00007f00f5023c01: add     $0x50,%rax
72 0x00007f00f5023c05: cmpq    $0x0,0x158(%r15)
73 0x00007f00f5023c10: jne     0x00007f00f5023c8d
74 0x00007f00f5023c16: mov     %rsp,-0x28(%rsp)
75 0x00007f00f5023c1b: sub     $0x80,%rsp
76 0x00007f00f5023c22: mov     %rax,0x78(%rsp)
77 0x00007f00f5023c27: mov     %rcx,0x70(%rsp)
78 0x00007f00f5023c2c: mov     %rdx,0x68(%rsp)
79 0x00007f00f5023c31: mov     %rbx,0x60(%rsp)
80 0x00007f00f5023c36: mov     %rbp,0x50(%rsp)
81 0x00007f00f5023c3b: mov     %rsi,0x48(%rsp)
82 0x00007f00f5023c40: mov     %rdi,0x40(%rsp)
83 0x00007f00f5023c45: mov     %r8,0x38(%rsp)
84 0x00007f00f5023c4a: mov     %r9,0x30(%rsp)

```

```

81 0x00007f00f5023c4f: mov    %r10,0x28(%rsp)
82 0x00007f00f5023c54: mov    %r11,0x20(%rsp)
83 0x00007f00f5023c59: mov    %r12,0x18(%rsp)
84 0x00007f00f5023c5e: mov    %r13,0x10(%rsp)
85 0x00007f00f5023c63: mov    %r14,0x8(%rsp)
86 0x00007f00f5023c68: mov    %r15, (%rsp)
87 0x00007f00f5023c6c: movabs $0x7f010b9187cc,%rdi
88 0x00007f00f5023c76: movabs $0x7f00f5023c16,%rsi
89 0x00007f00f5023c80: mov    %rsp,%rdx
90 0x00007f00f5023c83: and    $0xffffffffffffff0,%rsp
91 0x00007f00f5023c87: callq  0x00007f010b33b9fa
92 0x00007f00f5023c8c: hlt
93 0x00007f00f5023c8d: cmpq   $0x0,0x160(%r15)
94 0x00007f00f5023c98: jne    0x00007f00f5023d15
95 0x00007f00f5023c9e: mov    %rsp,-0x28(%rsp)
96 0x00007f00f5023ca3: sub    $0x80,%rsp
97 0x00007f00f5023caa: mov    %rax,0x78(%rsp)
98 0x00007f00f5023caf: mov    %rcx,0x70(%rsp)
99 0x00007f00f5023cb4: mov    %rdx,0x68(%rsp)
100 0x00007f00f5023cb9: mov    %rbx,0x60(%rsp)
101 0x00007f00f5023cbe: mov    %rbp,0x50(%rsp)
102 0x00007f00f5023cc3: mov    %rsi,0x48(%rsp)
103 0x00007f00f5023cc8: mov    %rdi,0x40(%rsp)
104 0x00007f00f5023ccd: mov    %r8,0x38(%rsp)
105 0x00007f00f5023cd2: mov    %r9,0x30(%rsp)
106 0x00007f00f5023cd7: mov    %r10,0x28(%rsp)
107 0x00007f00f5023cdc: mov    %r11,0x20(%rsp)
108 0x00007f00f5023ce1: mov    %r12,0x18(%rsp)
109 0x00007f00f5023ce6: mov    %r13,0x10(%rsp)
110 0x00007f00f5023ceb: mov    %r14,0x8(%rsp)
111 0x00007f00f5023cf0: mov    %r15, (%rsp)
112 0x00007f00f5023cf4: movabs $0x7f010b9187df,%rdi
113 0x00007f00f5023cfe: movabs $0x7f00f5023c9e,%rsi
114 0x00007f00f5023d08: mov    %rsp,%rdx
115 0x00007f00f5023d0b: and    $0xffffffffffffff0,%rsp
116 0x00007f00f5023d0f: callq  0x00007f010b33b9fa
117 0x00007f00f5023d14: hlt
118 0x00007f00f5023d15: add    0x158(%r15),%rax
119 0x00007f00f5023d1c: sub    0x160(%r15),%rax
120 0x00007f00f5023d23: add    $0x16000,%rax
121 0x00007f00f5023d2a: cmp    %rax,%rsp
122 0x00007f00f5023d2d: ja     0x00007f00f5023d3d
123 0x00007f00f5023d33: pop    %rax
124 0x00007f00f5023d34: mov    %r13,%rsp
125 0x00007f00f5023d37: push   %rax
126 0x00007f00f5023d38: jmpq   0x00007f00f5005280
127 0x00007f00f5023d3d: pop    %rax
128 0x00007f00f5023d3e: lea    -0x8(%rsp,%rcx,8),%r14
129 0x00007f00f5023d43: test   %edx,%edx
130 0x00007f00f5023d45: jle    0x00007f00f5023d54
131 0x00007f00f5023d4b: pushq  $0x0
132 0x00007f00f5023d50: dec    %edx
133 0x00007f00f5023d52: jg     0x00007f00f5023d4b
134 0x00007f00f5023d54: push   %rax
135 0x00007f00f5023d55: push   %rbp
136 0x00007f00f5023d56: mov    %rsp,%rbp
137 0x00007f00f5023d59: push   %r13
138 0x00007f00f5023d5b: pushq  $0x0
139 0x00007f00f5023d60: mov    0x10(%rbx),%r13
140 0x00007f00f5023d64: lea    0x30(%r13),%r13
141 0x00007f00f5023d68: push   %rbx
142 0x00007f00f5023d69: mov    0x18(%rbx),%rdx
143 0x00007f00f5023d6d: test   %rdx,%rdx
144 0x00007f00f5023d70: je     0x00007f00f5023d7d
145 0x00007f00f5023d76: add    $0x128,%rdx
146 0x00007f00f5023d7d: push   %rdx
147 0x00007f00f5023d7e: mov    0x10(%rbx),%rdx
148 0x00007f00f5023d82: mov    0x8(%rdx),%rdx
149 0x00007f00f5023d86: mov    0x18(%rdx),%rdx
150 0x00007f00f5023d8a: push   %rdx
151 0x00007f00f5023d8b: push   %r14
152 0x00007f00f5023d8d: push   %r13
153 0x00007f00f5023d8f: pushq  $0x0
154 0x00007f00f5023d94: mov    %rsp, (%rsp)
155 0x00007f00f5023d98: mov    0x28(%rbx),%eax
156 0x00007f00f5023d9b: test   $0x100,%eax
157 0x00007f00f5023da0: je     0x00007f00f5023e1d
158 0x00007f00f5023da6: mov    %rsp,-0x28(%rsp)
159 0x00007f00f5023dab: sub    $0x80,%rsp
160 0x00007f00f5023db2: mov    %rax,0x78(%rsp)
161 0x00007f00f5023db7: mov    %rcx,0x70(%rsp)
162 0x00007f00f5023dbc: mov    %rdx,0x68(%rsp)
163 0x00007f00f5023dc1: mov    %rbx,0x60(%rsp)
164 0x00007f00f5023dc6: mov    %rbp,0x50(%rsp)
165 0x00007f00f5023dcb: mov    %rsi,0x48(%rsp)
166 0x00007f00f5023dd0: mov    %rdi,0x40(%rsp)

```

```

167 0x00007f00f5023dd5: mov    %r8,0x38(%rsp)
168 0x00007f00f5023dda: mov    %r9,0x30(%rsp)
169 0x00007f00f5023ddf: mov    %r10,0x28(%rsp)
170 0x00007f00f5023de4: mov    %r11,0x20(%rsp)
171 0x00007f00f5023de9: mov    %r12,0x18(%rsp)
172 0x00007f00f5023dee: mov    %r13,0x10(%rsp)
173 0x00007f00f5023df3: mov    %r14,0x8(%rsp)
174 0x00007f00f5023df8: mov    %r15, (%rsp)
175 0x00007f00f5023dfc: movabs $0x7f010b918b30,%rdi
176 0x00007f00f5023e06: movabs $0x7f00f5023da6,%rsi
177 0x00007f00f5023e10: mov    %rsp,%rdx
178 0x00007f00f5023e13: and    $0xfffffffffffff0,%rsp
179 0x00007f00f5023e17: callq  0x00007f010b33b9fa
180 0x00007f00f5023e1c: hlt
181 0x00007f00f5023e1d: test   $0x400,%eax
182 0x00007f00f5023e22: je     0x00007f00f5023e9f
183 0x00007f00f5023e28: mov    %rsp,-0x28(%rsp)
184 0x00007f00f5023e2d: sub    $0x80,%rsp
185 0x00007f00f5023e34: mov    %rax,0x78(%rsp)
186 0x00007f00f5023e39: mov    %rcx,0x70(%rsp)
187 0x00007f00f5023e3e: mov    %rdx,0x68(%rsp)
188 0x00007f00f5023e43: mov    %rbx,0x60(%rsp)
189 0x00007f00f5023e48: mov    %rbp,0x50(%rsp)
190 0x00007f00f5023e4d: mov    %rsi,0x48(%rsp)
191 0x00007f00f5023e52: mov    %rdi,0x40(%rsp)
192 0x00007f00f5023e57: mov    %r8,0x38(%rsp)
193 0x00007f00f5023e5c: mov    %r9,0x30(%rsp)
194 0x00007f00f5023e61: mov    %r10,0x28(%rsp)
195 0x00007f00f5023e66: mov    %r11,0x20(%rsp)
196 0x00007f00f5023e6b: mov    %r12,0x18(%rsp)
197 0x00007f00f5023e70: mov    %r13,0x10(%rsp)
198 0x00007f00f5023e75: mov    %r14,0x8(%rsp)
199 0x00007f00f5023e7a: mov    %r15, (%rsp)
200 0x00007f00f5023e7e: movabs $0x7f010b918980,%rdi
201 0x00007f00f5023e88: movabs $0x7f00f5023e28,%rsi
202 0x00007f00f5023e92: mov    %rsp,%rdx
203 0x00007f00f5023e95: and    $0xfffffffffffff0,%rsp
204 0x00007f00f5023e99: callq  0x00007f010b33b9fa
205 0x00007f00f5023e9e: hlt
206 0x00007f00f5023e9f: movb   $0x1,0x2bd(%r15)
207 0x00007f00f5023ea7: mov    -0x20(%rbp),%rax
208 0x00007f00f5023eab: test   %rax,%rax
209 0x00007f00f5023eae: je     0x00007f00f5023f3e
210 0x00007f00f5023eb4: mov    -0x4(%rax),%ecx
211 0x00007f00f5023eb7: test   %ecx,%ecx
212 0x00007f00f5023eb9: js     0x00007f00f5023f3e
213 0x00007f00f5023ebf: add    %rcx,%rax
214 0x00007f00f5023ec2: mov    0x8(%rax),%rcx
215 0x00007f00f5023ec6: sub    $0x2,%rcx
216 0x00007f00f5023eca: mov    0x10(%rax,%rcx,8),%rdx
217 0x00007f00f5023ecf: neg    %rdx
218 0x00007f00f5023ed2: mov    (%r14,%rdx,8),%rdx
219 0x00007f00f5023ed6: test   %rdx,%rdx
220 0x00007f00f5023ed9: jne    0x00007f00f5023ee6
221 0x00007f00f5023edb: orq    $0x1,0x18(%rax,%rcx,8)
222 0x00007f00f5023ee4: jmp    0x00007f00f5023f38
223 0x00007f00f5023ee6: mov    0x8(%rdx),%edx
224 0x00007f00f5023ee9: shl    $0x3,%rdx
225 0x00007f00f5023eed: xor    0x18(%rax,%rcx,8),%rdx
226 0x00007f00f5023ef2: test   $0xfffffffffffffc,%rdx
227 0x00007f00f5023ef9: je     0x00007f00f5023f38
228 0x00007f00f5023efb: test   $0x2,%rdx
229 0x00007f00f5023f02: jne    0x00007f00f5023f38
230 0x00007f00f5023f04: cmpq   $0x0,0x18(%rax,%rcx,8)
231 0x00007f00f5023f0d: je     0x00007f00f5023f33
232 0x00007f00f5023f0f: cmpq   $0x1,0x18(%rax,%rcx,8)
233 0x00007f00f5023f18: je     0x00007f00f5023f33
234 0x00007f00f5023f1a: xor    0x18(%rax,%rcx,8),%rdx
235 0x00007f00f5023f1f: test   $0xfffffffffffffc,%rdx
236 0x00007f00f5023f26: je     0x00007f00f5023f38
237 0x00007f00f5023f28: orq    $0x2,0x18(%rax,%rcx,8)
238 0x00007f00f5023f31: jmp    0x00007f00f5023f38
239 0x00007f00f5023f33: mov    %rdx,0x18(%rax,%rcx,8)
240 0x00007f00f5023f38: sub    $0x2,%rcx
241 0x00007f00f5023f3c: jns    0x00007f00f5023eca
242 0x00007f00f5023f3e: mov    0x18(%rbx),%rax
243 0x00007f00f5023f42: test   %rax,%rax
244 0x00007f00f5023f45: je     0x00007f00f5023f67
245 0x00007f00f5023f47: mov    0x104(%rax),%ecx
246 0x00007f00f5023f4d: add    $0x8,%ecx
247 0x00007f00f5023f50: mov    %ecx,0x104(%rax)
248 0x00007f00f5023f56: and    $0x3f8,%ecx
249 0x00007f00f5023f5c: je     0x00007f00f5024793
250 0x00007f00f5023f62: jmpq   0x00007f00f50241ea
251 0x00007f00f5023f67: mov    0x20(%rbx),%rax
252 0x00007f00f5023f6b: test   %rax,%rax

```

```
253 0x00007f00f5023f6e: jne 0x00007f00f50241d5
254 0x00007f00f5023f74: callq 0x00007f00f5023f7e
255 0x00007f00f5023f79: jmpq 0x00007f00f50241c8
256 0x00007f00f5023f7e: mov %rbx,%rsi
257 0x00007f00f5023f81: lea 0x8(%rsp),%rax
258 0x00007f00f5023f86: mov %r13,-0x38(%rbp)
259 0x00007f00f5023f8a: cmpq $0x0,-0x10(%rbp)
260 0x00007f00f5023f92: je 0x00007f00f502400f
261 0x00007f00f5023f98: mov %rsp,-0x28(%rsp)
262 0x00007f00f5023f9d: sub $0x80,%rsp
263 0x00007f00f5023fa4: mov %rax,0x78(%rsp)
264 0x00007f00f5023fa9: mov %rcx,0x70(%rsp)
265 0x00007f00f5023fae: mov %rdx,0x68(%rsp)
266 0x00007f00f5023fb3: mov %rbx,0x60(%rsp)
267 0x00007f00f5023fb8: mov %rbp,0x50(%rsp)
268 0x00007f00f5023fbd: mov %rsi,0x48(%rsp)
269 0x00007f00f5023fc2: mov %rdi,0x40(%rsp)
270 0x00007f00f5023fc7: mov %r8,0x38(%rsp)
271 0x00007f00f5023fcc: mov %r9,0x30(%rsp)
272 0x00007f00f5023fd1: mov %r10,0x28(%rsp)
273 0x00007f00f5023fd6: mov %r11,0x20(%rsp)
274 0x00007f00f5023fdb: mov %r12,0x18(%rsp)
275 0x00007f00f5023fe0: mov %r13,0x10(%rsp)
276 0x00007f00f5023fe5: mov %r14,0x8(%rsp)
277 0x00007f00f5023fea: mov %r15, (%rsp)
278 0x00007f00f5023fee: movabs $0x7f010b7e62e8,%rdi
279 0x00007f00f5023ff8: movabs $0x7f00f5023f98,%rsi
280 0x00007f00f5024002: mov %rsp,%rdx
281 0x00007f00f5024005: and $0xfffffffffffffffff0,%rsp
282 0x00007f00f5024009: callq 0x00007f010b33b9fa
283 0x00007f00f502400e: hlt
284 0x00007f00f502400f: push %r10
285 0x00007f00f5024011: cmp 0x16eea1c8(%rip),%r12 # 0x00007f010bf0e1e0
286 0x00007f00f5024018: je 0x00007f00f5024095
287 ;; call_VM_base: heap base corrupted?
288 0x00007f00f502401e: mov %rsp,-0x28(%rsp)
289 0x00007f00f5024023: sub $0x80,%rsp
290 0x00007f00f502402a: mov %rax,0x78(%rsp)
291 0x00007f00f502402f: mov %rcx,0x70(%rsp)
292 0x00007f00f5024034: mov %rdx,0x68(%rsp)
293 0x00007f00f5024039: mov %rbx,0x60(%rsp)
294 0x00007f00f502403e: mov %rbp,0x50(%rsp)
295 0x00007f00f5024043: mov %rsi,0x48(%rsp)
296 0x00007f00f5024048: mov %rdi,0x40(%rsp)
297 0x00007f00f502404d: mov %r8,0x38(%rsp)
298 0x00007f00f5024052: mov %r9,0x30(%rsp)
299 0x00007f00f5024057: mov %r10,0x28(%rsp)
300 0x00007f00f502405c: mov %r11,0x20(%rsp)
301 0x00007f00f5024061: mov %r12,0x18(%rsp)
302 0x00007f00f5024066: mov %r13,0x10(%rsp)
303 0x00007f00f502406b: mov %r14,0x8(%rsp)
304 0x00007f00f5024070: mov %r15, (%rsp)
305 0x00007f00f5024074: movabs $0x7f010b851078,%rdi
306 0x00007f00f502407e: movabs $0x7f00f502401e,%rsi
307 0x00007f00f5024088: mov %rsp,%rdx
308 0x00007f00f502408b: and $0xfffffffffffffffff0,%rsp
309 0x00007f00f502408f: callq 0x00007f010b33b9fa
310 0x00007f00f5024094: hlt
311 0x00007f00f5024095: pop %r10
312 0x00007f00f5024097: mov %r15,%rdi
313 0x00007f00f502409a: mov %rbp,0x210(%r15)
314 0x00007f00f50240a1: mov %rax,0x200(%r15)
315 0x00007f00f50240a8: test $0xf,%esp
316 0x00007f00f50240ae: je 0x00007f00f50240c6
317 0x00007f00f50240b4: sub $0x8,%rsp
318 0x00007f00f50240b8: callq 0x00007f010b185330
319 0x00007f00f50240bd: add $0x8,%rsp
320 0x00007f00f50240c1: jmpq 0x00007f00f50240cb
321 0x00007f00f50240c6: callq 0x00007f010b185330
322 0x00007f00f50240cb: push %rax
323 0x00007f00f50240cc: push %rdi
324 0x00007f00f50240cd: push %rsi
325 0x00007f00f50240ce: push %rdx
326 0x00007f00f50240cf: push %rcx
327 0x00007f00f50240d0: push %r8
328 0x00007f00f50240d2: push %r9
329 0x00007f00f50240d4: push %r10
330 0x00007f00f50240d6: mov %rsp,%r10
331 0x00007f00f50240d9: and $0xfffffffffffffffff0,%rsp
332 0x00007f00f50240dd: push %r10
333 0x00007f00f50240df: push %r11
334 0x00007f00f50240e1: mov $0x1,%edi
335 0x00007f00f50240e6: callq 0x00007f010bf232d0
336 0x00007f00f50240eb: pop %r11
337 0x00007f00f50240ed: pop %rsp
338 0x00007f00f50240ee: pop %r10
```

```
339 0x00007f00f50240f0: pop %r9
340 0x00007f00f50240f2: pop %r8
341 0x00007f00f50240f4: pop %rcx
342 0x00007f00f50240f5: pop %rdx
343 0x00007f00f50240f6: pop %rsi
344 0x00007f00f50240f7: pop %rdi
345 0x00007f00f50240f8: cmp %rax,%r15
346 0x00007f00f50240fb: je 0x00007f00f5024178
347 ;; MacroAssembler::call_VM_base: rdi not callee saved?
348 0x00007f00f5024101: mov %rsp,-0x28(%rsp)
349 0x00007f00f5024106: sub $0x80,%rsp
350 0x00007f00f502410d: mov %rax,0x78(%rsp)
351 0x00007f00f5024112: mov %rcx,0x70(%rsp)
352 0x00007f00f5024117: mov %rdx,0x68(%rsp)
353 0x00007f00f502411c: mov %rbx,0x60(%rsp)
354 0x00007f00f5024121: mov %rbp,0x50(%rsp)
355 0x00007f00f5024126: mov %rsi,0x48(%rsp)
356 0x00007f00f502412b: mov %rdi,0x40(%rsp)
357 0x00007f00f5024130: mov %r8,0x38(%rsp)
358 0x00007f00f5024135: mov %r9,0x30(%rsp)
359 0x00007f00f502413a: mov %r10,0x28(%rsp)
360 0x00007f00f502413f: mov %r11,0x20(%rsp)
361 0x00007f00f5024144: mov %r12,0x18(%rsp)
362 0x00007f00f5024149: mov %r13,0x10(%rsp)
363 0x00007f00f502414e: mov %r14,0x8(%rsp)
364 0x00007f00f5024153: mov %r15,0(%rsp)
365 0x00007f00f5024157: movabs $0x7f010b8511f0,%rdi
366 0x00007f00f5024161: movabs $0x7f00f5024101,%rsi
367 0x00007f00f502416b: mov %rsp,%rdx
368 0x00007f00f502416e: and $0xffffffffffffffff,%rsp
369 0x00007f00f5024172: callq 0x00007f010b33b9fa
370 0x00007f00f5024177: hlt
371 0x00007f00f5024178: pop %rax
372 0x00007f00f5024179: movabs $0x0,%r10
373 0x00007f00f5024183: mov %r10,0x200(%r15)
374 0x00007f00f502418a: movabs $0x0,%r10
375 0x00007f00f5024194: mov %r10,0x210(%r15)
376 0x00007f00f502419b: movabs $0x0,%r10
377 0x00007f00f50241a5: mov %r10,0x208(%r15)
378 0x00007f00f50241ac: cmpq $0x0,0x8(%r15)
379 0x00007f00f50241b4: je 0x00007f00f50241bf
380 0x00007f00f50241ba: jmpq 0x00007f00f5000420
381 0x00007f00f50241bf: mov -0x38(%rbp),%r13
382 0x00007f00f50241c3: mov -0x30(%rbp),%r14
383 0x00007f00f50241c7: retq
384 0x00007f00f50241c8: mov 0x20(%rbx),%rax
385 0x00007f00f50241cc: test %rax,%rax
386 0x00007f00f50241cf: je 0x00007f00f50241ea
387 0x00007f00f50241d5: mov 0x8(%rax),%ecx
388 0x00007f00f50241d8: add $0x8,%ecx
389 0x00007f00f50241db: mov %ecx,0x8(%rax)
390 0x00007f00f50241de: and $0x3f8,%ecx
391 0x00007f00f50241e4: je 0x00007f00f5024793
392 0x00007f00f50241ea: mov %eax,-0x1000(%rsp)
393 0x00007f00f50241f1: mov %eax,-0x2000(%rsp)
394 0x00007f00f50241f8: mov %eax,-0x3000(%rsp)
395 0x00007f00f50241ff: mov %eax,-0x4000(%rsp)
396 0x00007f00f5024206: mov %eax,-0x5000(%rsp)
397 0x00007f00f502420d: mov %eax,-0x6000(%rsp)
398 0x00007f00f5024214: mov %eax,-0x7000(%rsp)
399 0x00007f00f502421b: mov %eax,-0x8000(%rsp)
400 0x00007f00f5024222: mov %eax,-0x9000(%rsp)
401 0x00007f00f5024229: mov %eax,-0xa000(%rsp)
402 0x00007f00f5024230: mov %eax,-0xb000(%rsp)
403 0x00007f00f5024237: mov %eax,-0xc000(%rsp)
404 0x00007f00f502423e: mov %eax,-0xd000(%rsp)
405 0x00007f00f5024245: mov %eax,-0xe000(%rsp)
406 0x00007f00f502424c: mov %eax,-0xf000(%rsp)
407 0x00007f00f5024253: mov %eax,-0x10000(%rsp)
408 0x00007f00f502425a: mov %eax,-0x11000(%rsp)
409 0x00007f00f5024261: mov %eax,-0x12000(%rsp)
410 0x00007f00f5024268: mov %eax,-0x13000(%rsp)
411 0x00007f00f502426f: mov %eax,-0x14000(%rsp)
412 0x00007f00f5024276: mov %eax,-0x15000(%rsp)
413 0x00007f00f502427d: mov %eax,-0x16000(%rsp)
414 0x00007f00f5024284: movb $0x0,0x2bd(%r15)
415 0x00007f00f502428c: mov 0x28(%rbx),%eax
416 0x00007f00f502428f: test $0x20,%eax
417 0x00007f00f5024294: je 0x00007f00f5024311
418 0x00007f00f502429a: mov %rsp,-0x28(%rsp)
419 0x00007f00f502429f: sub $0x80,%rsp
420 0x00007f00f50242a6: mov %rax,0x78(%rsp)
421 0x00007f00f50242ab: mov %rcx,0x70(%rsp)
422 0x00007f00f50242b0: mov %rdx,0x68(%rsp)
423 0x00007f00f50242b5: mov %rbx,0x60(%rsp)
424 0x00007f00f50242ba: mov %rbp,0x50(%rsp)
```

```

425 0x00007f00f50242bf: mov    %rsi,0x48(%rsp)
426 0x00007f00f50242c4: mov    %rdi,0x40(%rsp)
427 0x00007f00f50242c9: mov    %r8,0x38(%rsp)
428 0x00007f00f50242ce: mov    %r9,0x30(%rsp)
429 0x00007f00f50242d3: mov    %r10,0x28(%rsp)
430 0x00007f00f50242d8: mov    %r11,0x20(%rsp)
431 0x00007f00f50242dd: mov    %r12,0x18(%rsp)
432 0x00007f00f50242e2: mov    %r13,0x10(%rsp)
433 0x00007f00f50242e7: mov    %r14,0x8(%rsp)
434 0x00007f00f50242ec: mov    %r15, (%rsp)
435 0x00007f00f50242f0: movabs $0x7f010b9189b0,%rdi
436 0x00007f00f50242fa: movabs $0x7f00f502429a,%rsi
437 0x00007f00f5024304: mov    %rsp,%rdx
438 0x00007f00f5024307: and    $0xffffffffffffff0,%rsp
439 0x00007f00f502430b: callq  0x00007f010b33b9fa
440 0x00007f00f5024310: hlt
441 0x00007f00f5024311: mov    -0x40(%rbp),%rax
442 0x00007f00f5024315: cmp    %rsp,%rax
443 0x00007f00f5024318: je     0x00007f00f5024395
444 0x00007f00f502431e: mov    %rsp,-0x28(%rsp)
445 0x00007f00f5024323: sub    $0x80,%rsp
446 0x00007f00f502432a: mov    %rax,0x78(%rsp)
447 0x00007f00f502432f: mov    %rcx,0x70(%rsp)
448 0x00007f00f5024334: mov    %rdx,0x68(%rsp)
449 0x00007f00f5024339: mov    %rbx,0x60(%rsp)
450 0x00007f00f502433e: mov    %rbp,0x50(%rsp)
451 0x00007f00f5024343: mov    %rsi,0x48(%rsp)
452 0x00007f00f5024348: mov    %rdi,0x40(%rsp)
453 0x00007f00f502434d: mov    %r8,0x38(%rsp)
454 0x00007f00f5024352: mov    %r9,0x30(%rsp)
455 0x00007f00f5024357: mov    %r10,0x28(%rsp)
456 0x00007f00f502435c: mov    %r11,0x20(%rsp)
457 0x00007f00f5024361: mov    %r12,0x18(%rsp)
458 0x00007f00f5024366: mov    %r13,0x10(%rsp)
459 0x00007f00f502436b: mov    %r14,0x8(%rsp)
460 0x00007f00f5024370: mov    %r15, (%rsp)
461 0x00007f00f5024374: movabs $0x7f010b9189d0,%rdi
462 0x00007f00f502437e: movabs $0x7f00f502431e,%rsi
463 0x00007f00f5024388: mov    %rsp,%rdx
464 0x00007f00f502438b: and    $0xffffffffffffff0,%rsp
465 0x00007f00f502438f: callq  0x00007f010b33b9fa
466 0x00007f00f5024394: hlt
467 0x00007f00f5024395: cmpb   $0x0,0x16eb729e(%rip)    # 0x00007f010bedb63a
468 0x00007f00f502439c: je     0x00007f00f5024451
469 0x00007f00f50243a2: mov    -0x18(%rbp),%rsi
470 0x00007f00f50243a6: mov    %r15,%rdi
471 0x00007f00f50243a9: cmpq   $0x0,-0x10(%rbp)
472 0x00007f00f50243b1: je     0x00007f00f502442e
473 0x00007f00f50243b7: mov    %rsp,-0x28(%rsp)
474 0x00007f00f50243bc: sub    $0x80,%rsp
475 0x00007f00f50243c3: mov    %rax,0x78(%rsp)
476 0x00007f00f50243c8: mov    %rcx,0x70(%rsp)
477 0x00007f00f50243cd: mov    %rdx,0x68(%rsp)
478 0x00007f00f50243d2: mov    %rbx,0x60(%rsp)
479 0x00007f00f50243d7: mov    %rbp,0x50(%rsp)
480 0x00007f00f50243dc: mov    %rsi,0x48(%rsp)
481 0x00007f00f50243e1: mov    %rdi,0x40(%rsp)
482 0x00007f00f50243e6: mov    %r8,0x38(%rsp)
483 0x00007f00f50243eb: mov    %r9,0x30(%rsp)
484 0x00007f00f50243f0: mov    %r10,0x28(%rsp)
485 0x00007f00f50243f5: mov    %r11,0x20(%rsp)
486 0x00007f00f50243fa: mov    %r12,0x18(%rsp)
487 0x00007f00f50243ff: mov    %r13,0x10(%rsp)
488 0x00007f00f5024404: mov    %r14,0x8(%rsp)
489 0x00007f00f5024409: mov    %r15, (%rsp)
490 0x00007f00f502440d: movabs $0x7f010b7e62e8,%rdi
491 0x00007f00f5024417: movabs $0x7f00f50243b7,%rsi
492 0x00007f00f5024421: mov    %rsp,%rdx
493 0x00007f00f5024424: and    $0xffffffffffffff0,%rsp
494 0x00007f00f5024428: callq  0x00007f010b33b9fa
495 0x00007f00f502442d: hlt
496 0x00007f00f502442e: test   $0xf,%esp
497 0x00007f00f5024434: je     0x00007f00f502444c
498 0x00007f00f502443a: sub    $0x8,%rsp
499 0x00007f00f502443e: callq  0x00007f010b4e8092
500 0x00007f00f5024443: add    $0x8,%rsp
501 0x00007f00f5024447: jmpq   0x00007f00f5024451
502 0x00007f00f502444c: callq  0x00007f010b4e8092
503 0x00007f00f5024451: movzbl 0x0(%r13),%ebx
504 0x00007f00f5024456: movabs $0x7f010befeae0,%r10
505 0x00007f00f5024460: jmpq   *(%r10,%rbx,8)
506 0x00007f00f5024464: callq  0x00007f00f502446e
507 0x00007f00f5024469: jmpq   0x00007f00f5024465
508 0x00007f00f502446e: lea    0x8(%rsp),%rax
509 0x00007f00f5024473: mov    %r13,-0x38(%rbp)
510 0x00007f00f5024477: cmpq   $0x0,-0x10(%rbp)

```

```

511 0x00007f00f502447f: je 0x00007f00f50244fc
512 0x00007f00f5024485: mov %rsp, -0x28(%rsp)
513 0x00007f00f502448a: sub $0x80, %rsp
514 0x00007f00f5024491: mov %rax, 0x78(%rsp)
515 0x00007f00f5024496: mov %rcx, 0x70(%rsp)
516 0x00007f00f502449b: mov %rdx, 0x68(%rsp)
517 0x00007f00f50244a0: mov %rbx, 0x60(%rsp)
518 0x00007f00f50244a5: mov %rbp, 0x50(%rsp)
519 0x00007f00f50244aa: mov %rsi, 0x48(%rsp)
520 0x00007f00f50244af: mov %rdi, 0x40(%rsp)
521 0x00007f00f50244b4: mov %r8, 0x38(%rsp)
522 0x00007f00f50244b9: mov %r9, 0x30(%rsp)
523 0x00007f00f50244be: mov %r10, 0x28(%rsp)
524 0x00007f00f50244c3: mov %r11, 0x20(%rsp)
525 0x00007f00f50244c8: mov %r12, 0x18(%rsp)
526 0x00007f00f50244cd: mov %r13, 0x10(%rsp)
527 0x00007f00f50244d2: mov %r14, 0x8(%rsp)
528 0x00007f00f50244d7: mov %r15, (%rsp)
529 0x00007f00f50244db: movabs $0x7f010b7e62e8, %rdi
530 0x00007f00f50244e5: movabs $0x7f00f5024485, %rsi
531 0x00007f00f50244ef: mov %rsp, %rdx
532 0x00007f00f50244f2: and $0xfffffffffffff0, %rsp
533 0x00007f00f50244f6: callq 0x00007f010b33b9fa
534 0x00007f00f50244fb: hlt
535 0x00007f00f50244fc: push %r10
536 0x00007f00f50244fe: cmp 0x16ee9cdb(%rip), %r12 # 0x00007f010bf0e1e0
537 0x00007f00f5024505: je 0x00007f00f5024582
538 ;; call_VM_base: heap base corrupted?
539 0x00007f00f502450b: mov %rsp, -0x28(%rsp)
540 0x00007f00f5024510: sub $0x80, %rsp
541 0x00007f00f5024517: mov %rax, 0x78(%rsp)
542 0x00007f00f502451c: mov %rcx, 0x70(%rsp)
543 0x00007f00f5024521: mov %rdx, 0x68(%rsp)
544 0x00007f00f5024526: mov %rbx, 0x60(%rsp)
545 0x00007f00f502452b: mov %rbp, 0x50(%rsp)
546 0x00007f00f5024530: mov %rsi, 0x48(%rsp)
547 0x00007f00f5024535: mov %rdi, 0x40(%rsp)
548 0x00007f00f502453a: mov %r8, 0x38(%rsp)
549 0x00007f00f502453f: mov %r9, 0x30(%rsp)
550 0x00007f00f5024544: mov %r10, 0x28(%rsp)
551 0x00007f00f5024549: mov %r11, 0x20(%rsp)
552 0x00007f00f502454e: mov %r12, 0x18(%rsp)
553 0x00007f00f5024553: mov %r13, 0x10(%rsp)
554 0x00007f00f5024558: mov %r14, 0x8(%rsp)
555 0x00007f00f502455d: mov %r15, (%rsp)
556 0x00007f00f5024561: movabs $0x7f010b851078, %rdi
557 0x00007f00f502456b: movabs $0x7f00f502450b, %rsi
558 0x00007f00f5024575: mov %rsp, %rdx
559 0x00007f00f5024578: and $0xfffffffffffff0, %rsp
560 0x00007f00f502457c: callq 0x00007f010b33b9fa
561 0x00007f00f5024581: hlt
562 0x00007f00f5024582: pop %r10
563 0x00007f00f5024584: mov %r15, %rdi
564 0x00007f00f5024587: mov %rbp, 0x210(%r15)
565 0x00007f00f502458e: mov %rax, 0x200(%r15)
566 0x00007f00f5024595: test $0xf, %esp
567 0x00007f00f502459b: je 0x00007f00f50245b3
568 0x00007f00f50245a1: sub $0x8, %rsp
569 0x00007f00f50245a5: callq 0x00007f010b1848ea
570 0x00007f00f50245aa: add $0x8, %rsp
571 0x00007f00f50245ae: jmpq 0x00007f00f50245b8
572 0x00007f00f50245b3: callq 0x00007f010b1848ea
573 0x00007f00f50245b8: push %rax
574 0x00007f00f50245b9: push %rdi
575 0x00007f00f50245ba: push %rsi
576 0x00007f00f50245bb: push %rdx
577 0x00007f00f50245bc: push %rcx
578 0x00007f00f50245bd: push %r8
579 0x00007f00f50245bf: push %r9
580 0x00007f00f50245c1: push %r10
581 0x00007f00f50245c3: mov %rsp, %r10
582 0x00007f00f50245c6: and $0xfffffffffffff0, %rsp
583 0x00007f00f50245ca: push %r10
584 0x00007f00f50245cc: push %r11
585 0x00007f00f50245ce: mov $0x1, %edi
586 0x00007f00f50245d3: callq 0x00007f010bf232d0
587 0x00007f00f50245d8: pop %r11
588 0x00007f00f50245da: pop %rsp
589 0x00007f00f50245db: pop %r10
590 0x00007f00f50245dd: pop %r9
591 0x00007f00f50245df: pop %r8
592 0x00007f00f50245e1: pop %rcx
593 0x00007f00f50245e2: pop %rdx
594 0x00007f00f50245e3: pop %rsi
595 0x00007f00f50245e4: pop %rdi
596 0x00007f00f50245e5: cmp %rax, %r15

```

```

597 0x00007f00f50245e8: je      0x00007f00f5024665
598 ;; MacroAssembler::call_VM_base: rdi not callee saved?
599 0x00007f00f50245ee: mov     %rsp, -0x28(%rsp)
600 0x00007f00f50245f3: sub     $0x80, %rsp
601 0x00007f00f50245fa: mov     %rax, 0x78(%rsp)
602 0x00007f00f50245ff: mov     %rcx, 0x70(%rsp)
603 0x00007f00f5024604: mov     %rdx, 0x68(%rsp)
604 0x00007f00f5024609: mov     %rbx, 0x60(%rsp)
605 0x00007f00f502460e: mov     %rbp, 0x50(%rsp)
606 0x00007f00f5024613: mov     %rsi, 0x48(%rsp)
607 0x00007f00f5024618: mov     %rdi, 0x40(%rsp)
608 0x00007f00f502461d: mov     %r8, 0x38(%rsp)
609 0x00007f00f5024622: mov     %r9, 0x30(%rsp)
610 0x00007f00f5024627: mov     %r10, 0x28(%rsp)
611 0x00007f00f502462c: mov     %r11, 0x20(%rsp)
612 0x00007f00f5024631: mov     %r12, 0x18(%rsp)
613 0x00007f00f5024636: mov     %r13, 0x10(%rsp)
614 0x00007f00f502463b: mov     %r14, 0x8(%rsp)
615 0x00007f00f5024640: mov     %r15, (%rsp)
616 0x00007f00f5024644: movabs  $0x7f010b8511f0, %rdi
617 0x00007f00f502464e: movabs  $0x7f00f50245ee, %rsi
618 0x00007f00f5024658: mov     %rsp, %rdx
619 0x00007f00f502465b: and     $0xfffffffffffffffff0, %rsp
620 0x00007f00f502465f: callq   0x00007f010b33b9fa
621 0x00007f00f5024664: hlt
622 0x00007f00f5024665: pop     %rax
623 0x00007f00f5024666: movabs  $0x0, %r10
624 0x00007f00f5024670: mov     %r10, 0x200(%r15)
625 0x00007f00f5024677: movabs  $0x0, %r10
626 0x00007f00f5024681: mov     %r10, 0x210(%r15)
627 0x00007f00f5024688: movabs  $0x0, %r10
628 0x00007f00f5024692: mov     %r10, 0x208(%r15)
629 0x00007f00f5024699: cmpq    $0x0, 0x8(%r15)
630 0x00007f00f50246a1: je      0x00007f00f50246ac
631 0x00007f00f50246a7: jmpq    0x00007f00f5000420
632 0x00007f00f50246ac: mov     -0x38(%rbp), %r13
633 0x00007f00f50246b0: mov     -0x30(%rbp), %r14
634 0x00007f00f50246b4: retq
635 0x00007f00f50246b5: push    %rax
636 0x00007f00f50246b6: push    %rbx
637 0x00007f00f50246b7: mov     -0x18(%rbp), %rbx
638 0x00007f00f50246bb: mov     0x18(%rbx), %rax
639 0x00007f00f50246bf: test    %rax, %rax
640 0x00007f00f50246c2: je      0x00007f00f5024784
641 0x00007f00f50246c8: mov     %r13, %rsi
642 0x00007f00f50246cb: mov     %rbx, %rdi
643 0x00007f00f50246ce: cmpq    $0x0, -0x10(%rbp)
644 0x00007f00f50246d6: je      0x00007f00f5024753
645 0x00007f00f50246dc: mov     %rsp, -0x28(%rsp)
646 0x00007f00f50246e1: sub     $0x80, %rsp
647 0x00007f00f50246e8: mov     %rax, 0x78(%rsp)
648 0x00007f00f50246ed: mov     %rcx, 0x70(%rsp)
649 0x00007f00f50246f2: mov     %rdx, 0x68(%rsp)
650 0x00007f00f50246f7: mov     %rbx, 0x60(%rsp)
651 0x00007f00f50246fc: mov     %rbp, 0x50(%rsp)
652 0x00007f00f5024701: mov     %rsi, 0x48(%rsp)
653 0x00007f00f5024706: mov     %rdi, 0x40(%rsp)
654 0x00007f00f502470b: mov     %r8, 0x38(%rsp)
655 0x00007f00f5024710: mov     %r9, 0x30(%rsp)
656 0x00007f00f5024715: mov     %r10, 0x28(%rsp)
657 0x00007f00f502471a: mov     %r11, 0x20(%rsp)
658 0x00007f00f502471f: mov     %r12, 0x18(%rsp)
659 0x00007f00f5024724: mov     %r13, 0x10(%rsp)
660 0x00007f00f5024729: mov     %r14, 0x8(%rsp)
661 0x00007f00f502472e: mov     %r15, (%rsp)
662 0x00007f00f5024732: movabs  $0x7f010b7e62e8, %rdi
663 0x00007f00f502473c: movabs  $0x7f00f50246dc, %rsi
664 0x00007f00f5024746: mov     %rsp, %rdx
665 0x00007f00f5024749: and     $0xfffffffffffffffff0, %rsp
666 0x00007f00f502474d: callq   0x00007f010b33b9fa
667 0x00007f00f5024752: hlt
668 0x00007f00f5024753: test    $0xf, %esp
669 0x00007f00f5024759: je      0x00007f00f5024771
670 0x00007f00f502475f: sub     $0x8, %rsp
671 0x00007f00f5024763: callq   0x00007f010b184742
672 0x00007f00f5024768: add     $0x8, %rsp
673 0x00007f00f502476c: jmpq    0x00007f00f5024776
674 0x00007f00f5024771: callq   0x00007f010b184742
675 0x00007f00f5024776: mov     0x18(%rbx), %rbx
676 0x00007f00f502477a: add     $0x128, %rbx
677 0x00007f00f5024781: add     %rbx, %rax
678 0x00007f00f5024784: mov     %rax, -0x20(%rbp)
679 0x00007f00f5024788: pop     %rbx
680 0x00007f00f5024789: pop     %rax
681 0x00007f00f502478a: mov     -0x18(%rbp), %rbx
682 0x00007f00f502478e: jmpq    0x00007f00f50241ea

```



```

683 0x00007f00f5024793: mov    $0x0,%esi
684 0x00007f00f5024798: callq 0x00007f00f50247a2
685 0x00007f00f502479d: jmpq   0x00007f00f50249e9
686 0x00007f00f50247a2: lea    0x8(%rsp),%rax
687 0x00007f00f50247a7: mov    %r13,-0x38(%rbp)
688 0x00007f00f50247ab: cmpq   $0x0,-0x10(%rbp)
689 0x00007f00f50247b3: je     0x00007f00f5024830
690 0x00007f00f50247b9: mov    %rsp,-0x28(%rsp)
691 0x00007f00f50247be: sub    $0x80,%rsp
692 0x00007f00f50247c5: mov    %rax,0x78(%rsp)
693 0x00007f00f50247ca: mov    %rcx,0x70(%rsp)
694 0x00007f00f50247cf: mov    %rdx,0x68(%rsp)
695 0x00007f00f50247d4: mov    %rbx,0x60(%rsp)
696 0x00007f00f50247d9: mov    %rbp,0x50(%rsp)
697 0x00007f00f50247de: mov    %rsi,0x48(%rsp)
698 0x00007f00f50247e3: mov    %rdi,0x40(%rsp)
699 0x00007f00f50247e8: mov    %r8,0x38(%rsp)
700 0x00007f00f50247ed: mov    %r9,0x30(%rsp)
701 0x00007f00f50247f2: mov    %r10,0x28(%rsp)
702 0x00007f00f50247f7: mov    %r11,0x20(%rsp)
703 0x00007f00f50247fc: mov    %r12,0x18(%rsp)
704 0x00007f00f5024801: mov    %r13,0x10(%rsp)
705 0x00007f00f5024806: mov    %r14,0x8(%rsp)
706 0x00007f00f502480b: mov    %r15,0(%rsp)
707 0x00007f00f502480f: movabs $0x7f010b7e62e8,%rdi
708 0x00007f00f5024819: movabs $0x7f00f50247b9,%rsi
709 0x00007f00f5024823: mov    %rsp,%rdx
710 0x00007f00f5024826: and    $0xfffffffffffffff0,%rsp
711 0x00007f00f502482a: callq 0x00007f010b33b9fa
712 0x00007f00f502482f: hlt
713 0x00007f00f5024830: push   %r10
714 0x00007f00f5024832: cmp    0x16ee99a7(%rip),%r12    # 0x00007f010bf0e1e0
715 0x00007f00f5024839: je     0x00007f00f50248b6
716 ;; call_VM_base: heap base corrupted?
717 0x00007f00f502483f: mov    %rsp,-0x28(%rsp)
718 0x00007f00f5024844: sub    $0x80,%rsp
719 0x00007f00f502484b: mov    %rax,0x78(%rsp)
720 0x00007f00f5024850: mov    %rcx,0x70(%rsp)
721 0x00007f00f5024855: mov    %rdx,0x68(%rsp)
722 0x00007f00f502485a: mov    %rbx,0x60(%rsp)
723 0x00007f00f502485f: mov    %rbp,0x50(%rsp)
724 0x00007f00f5024864: mov    %rsi,0x48(%rsp)
725 0x00007f00f5024869: mov    %rdi,0x40(%rsp)
726 0x00007f00f502486e: mov    %r8,0x38(%rsp)
727 0x00007f00f5024873: mov    %r9,0x30(%rsp)
728 0x00007f00f5024878: mov    %r10,0x28(%rsp)
729 0x00007f00f502487d: mov    %r11,0x20(%rsp)
730 0x00007f00f5024882: mov    %r12,0x18(%rsp)
731 0x00007f00f5024887: mov    %r13,0x10(%rsp)
732 0x00007f00f502488c: mov    %r14,0x8(%rsp)
733 0x00007f00f5024891: mov    %r15,0(%rsp)
734 0x00007f00f5024895: movabs $0x7f010b851078,%rdi
735 0x00007f00f502489f: movabs $0x7f00f502483f,%rsi
736 0x00007f00f50248a9: mov    %rsp,%rdx
737 0x00007f00f50248ac: and    $0xfffffffffffffff0,%rsp
738 0x00007f00f50248b0: callq 0x00007f010b33b9fa
739 0x00007f00f50248b5: hlt
740 0x00007f00f50248b6: pop     %r10
741 0x00007f00f50248b8: mov    %r15,%rdi
742 0x00007f00f50248bb: mov    %rbp,0x210(%r15)
743 0x00007f00f50248c2: mov    %rax,0x200(%r15)
744 0x00007f00f50248c9: test   $0xf,%esp
745 0x00007f00f50248cf: je     0x00007f00f50248e7
746 0x00007f00f50248d5: sub    $0x8,%rsp
747 0x00007f00f50248d9: callq 0x00007f010b18415a
748 0x00007f00f50248de: add    $0x8,%rsp
749 0x00007f00f50248e2: jmpq   0x00007f00f50248ec
750 0x00007f00f50248e7: callq 0x00007f010b18415a
751 0x00007f00f50248ec: push   %rax
752 0x00007f00f50248ed: push   %rdi
753 0x00007f00f50248ee: push   %rsi
754 0x00007f00f50248ef: push   %rdx
755 0x00007f00f50248f0: push   %rcx
756 0x00007f00f50248f1: push   %r8
757 0x00007f00f50248f3: push   %r9
758 0x00007f00f50248f5: push   %r10
759 0x00007f00f50248f7: mov    %rsp,%r10
760 0x00007f00f50248fa: and    $0xfffffffffffffff0,%rsp
761 0x00007f00f50248fe: push   %r10
762 0x00007f00f5024900: push   %r11
763 0x00007f00f5024902: mov    $0x1,%edi
764 0x00007f00f5024907: callq 0x00007f010bf232d0
765 0x00007f00f502490c: pop     %r11
766 0x00007f00f502490e: pop     %rsp
767 0x00007f00f502490f: pop     %r10
768 0x00007f00f5024911: pop     %r9

```

```
769 0x00007f00f5024913: pop    %r8
770 0x00007f00f5024915: pop    %rcx
771 0x00007f00f5024916: pop    %rdx
772 0x00007f00f5024917: pop    %rsi
773 0x00007f00f5024918: pop    %rdi
774 0x00007f00f5024919: cmp    %rax,%r15
775 0x00007f00f502491c: je     0x00007f00f5024999
776 ; MacroAssembler::call_VM_base: rdi not callee saved?
777 0x00007f00f5024922: mov    %rsp,-0x28(%rsp)
778 0x00007f00f5024927: sub    $0x80,%rsp
779 0x00007f00f502492e: mov    %rax,0x78(%rsp)
780 0x00007f00f5024933: mov    %rcx,0x70(%rsp)
781 0x00007f00f5024938: mov    %rdx,0x68(%rsp)
782 0x00007f00f502493d: mov    %rbx,0x60(%rsp)
783 0x00007f00f5024942: mov    %rbp,0x50(%rsp)
784 0x00007f00f5024947: mov    %rsi,0x48(%rsp)
785 0x00007f00f502494c: mov    %rdi,0x40(%rsp)
786 0x00007f00f5024951: mov    %r8,0x38(%rsp)
787 0x00007f00f5024956: mov    %r9,0x30(%rsp)
788 0x00007f00f502495b: mov    %r10,0x28(%rsp)
789 0x00007f00f5024960: mov    %r11,0x20(%rsp)
790 0x00007f00f5024965: mov    %r12,0x18(%rsp)
791 0x00007f00f502496a: mov    %r13,0x10(%rsp)
792 0x00007f00f502496f: mov    %r14,0x8(%rsp)
793 0x00007f00f5024974: mov    %r15,0(%rsp)
794 0x00007f00f5024978: movabs $0x7f010b8511f0,%rdi
795 0x00007f00f5024982: movabs $0x7f00f5024922,%rsi
796 0x00007f00f502498c: mov    %rsp,%rdx
797 0x00007f00f502498f: and    $0xfffffffffffff0,%rsp
798 0x00007f00f5024993: callq  0x00007f010b33b9fa
799 0x00007f00f5024998: hlt
800 0x00007f00f5024999: pop    %rax
801 0x00007f00f502499a: movabs $0x0,%r10
802 0x00007f00f50249a4: mov    %r10,0x200(%r15)
803 0x00007f00f50249ab: movabs $0x0,%r10
804 0x00007f00f50249b5: mov    %r10,0x210(%r15)
805 0x00007f00f50249bc: movabs $0x0,%r10
806 0x00007f00f50249c6: mov    %r10,0x208(%r15)
807 0x00007f00f50249cd: cmpq   $0x0,0x8(%r15)
808 0x00007f00f50249d5: je     0x00007f00f50249e0
809 0x00007f00f50249db: jmpq   0x00007f00f5000420
810 0x00007f00f50249e0: mov    -0x38(%rbp),%r13
811 0x00007f00f50249e4: mov    -0x30(%rbp),%r14
812 0x00007f00f50249e8: retq
813 0x00007f00f50249e9: mov    -0x18(%rbp),%rbx
814 0x00007f00f50249ed: jmpq   0x00007f00f50241ea
815 0x00007f00f50249f2: nop
816 0x00007f00f50249f3: nop
817 0x00007f00f50249f4: nop
818 0x00007f00f50249f5: nop
819 0x00007f00f50249f6: nop
820 0x00007f00f50249f7: nop
821 0x00007f00f50249f8: int3
822 0x00007f00f50249f9: int3
823 0x00007f00f50249fa: int3
824 0x00007f00f50249fb: int3
825 0x00007f00f50249fc: int3
826 0x00007f00f50249fd: int3
827 0x00007f00f50249fe: int3
828 0x00007f00f50249ff: int3
829
830 -----
831
```