

## [PATCH 3/3] x86, kvm: make register\_address\_increment and JMP\_REL static inlines

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-02/msg09277.html>

---

- *From:* Harvey Harrison <[harvey.harrison@xxxxxxxxxx](mailto:harvey.harrison@xxxxxxxxxx)>
  - *Date:* Mon, 18 Feb 2008 11:12:51 -0800
- 

Change to jmp\_rel now that it is a function.

Signed-off-by: Harvey Harrison <[harvey.harrison@xxxxxxxxxx](mailto:harvey.harrison@xxxxxxxxxx)>

---

arch/x86/kvm/x86\_emulate.c | 58 ++++++-----  
1 files changed, 27 insertions(+), 31 deletions(-)

diff --git a/arch/x86/kvm/x86\_emulate.c b/arch/x86/kvm/x86\_emulate.c

index 5b2cc50..f5508d2 100644

--- a/arch/x86/kvm/x86\_emulate.c

+++ b/arch/x86/kvm/x86\_emulate.c

```
@@ -447,23 +447,19 @@ register_address(struct decode_cache *c, unsigned long base, unsigned long reg)
return base + address_mask(c, reg);
}
```

```
+#define register_address_increment(reg, inc) \
```

```
- do { \
```

```
- /* signed type ensures sign extension to long */ \
```

```
- int _inc = (inc); \
```

```
- if (c->ad_bytes == sizeof(unsigned long)) \
```

```
- (reg) += _inc; \
```

```
- else \
```

```
- (reg) = ((reg) & \
```

```
- ~ad_mask(c)) | \
```

```
- (((reg) + _inc) & \
```

```
- ad_mask(c)); \
```

```
- } while (0)
```

```
+static inline void
```

```
+register_address_increment(struct decode_cache *c, unsigned long *reg, int inc)
```

```
+{
```

```
+ if (c->ad_bytes == sizeof(unsigned long))
```

```
+ *reg += inc;
```

```
+ else
```

```
+ *reg = (*reg & ~ad_mask(c)) | ((*reg + inc) & ad_mask(c));
```

```
+}
```

[PATCH 3/3] x86, kvm: make register\_address\_increment and JMP\_REL static inlines

```

-#define JMP_REL(rel) \
- do { \
- register_address_increment(c->eip, rel); \
- } while (0)
+static inline void jmp_rel(struct decode_cache *c, int rel)
+{
+ register_address_increment(c, &c->eip, rel);
+}

static int do_fetch_insn_byte(struct x86_emulate_ctxt *ctxt,
struct x86_emulate_ops *ops,
@@ -999,7 +995,7 @@ static inline void emulate_push(struct x86_emulate_ctxt *ctxt)
c->dst.type = OP_MEM;
c->dst.bytes = c->op_bytes;
c->dst.val = c->src.val;
- register_address_increment(c->regs[VCPU_REGS_RSP], -c->op_bytes);
+ register_address_increment(c, &c->regs[VCPU_REGS_RSP], -c->op_bytes);
c->dst.ptr = (void *) register_address(c, ctxt->ss_base,
c->regs[VCPU_REGS_RSP]);
}
@@ -1016,7 +1012,7 @@ static inline int emulate_grp1a(struct x86_emulate_ctxt *ctxt,
if (rc != 0)
return rc;

- register_address_increment(c->regs[VCPU_REGS_RSP], c->dst.bytes);
+ register_address_increment(c, &c->regs[VCPU_REGS_RSP], c->dst.bytes);

return 0;
}
@@ -1127,7 +1123,7 @@ static inline int emulate_grp45(struct x86_emulate_ctxt *ctxt,
if (rc != 0)
return rc;
```