

Shiro 持久化 Session

作者: [Single](#)

原文链接: <https://ld246.com/article/1498702146290>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Session序列化反序列化工具

```
import org.apache.shiro.codec.Base64;
import org.apache.shiro.session.Session;

import java.io.ByteArrayInputStream;
import java.io.ByteArrayOutputStream;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;

public class SerializableUtils {

    public static String serialize(Session session) {
        try {
            ByteArrayOutputStream bos = new ByteArrayOutputStream();
            ObjectOutputStream oos = new ObjectOutputStream(bos);
            oos.writeObject(session);
            return Base64.encodeToString(bos.toByteArray());
        } catch (Exception e) {
            throw new RuntimeException("serialize session error", e);
        }
    }

    public static Session deserialize(String sessionStr) {
        try {
            ByteArrayInputStream bis = new ByteArrayInputStream(
                Base64.decode(sessionStr));
            ObjectInputStream ois = new ObjectInputStream(bis);
            return (Session) ois.readObject();
        } catch (Exception e) {
            throw new RuntimeException("deserialize session error", e);
        }
    }
}
```

持久化Session

```
import java.io.Serializable;
import java.util.List;

import org.apache.shiro.session.Session;
import org.apache.shiro.session.mgt.ValidatingSession;
import org.apache.shiro.session.mgt.eis.EnterpriseCacheSessionDAO;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.jdbc.core.JdbcTemplate;

public class JdbcSessionDao extends EnterpriseCacheSessionDAO {

    @Autowired
    private JdbcTemplate jdbcTemplate = null;

    @Override
```

```

protected Serializable doCreate(Session session) {
    Serializable sessionId = generateSessionId(session);
    assignSessionId(session, sessionId);
    String sql = "insert into sessions(id, session) values(?,?)";
    jdbcTemplate.update(sql, sessionId,
        SerializableUtils.serialize(session));
    return session.getId();
}

@Override
protected Session doReadSession(Serializable sessionId) {
    String sql = "select session from sessions where id=?";
    List<String> sessionStrList = jdbcTemplate.queryForList(sql,
        String.class, sessionId);
    if (sessionStrList.size() == 0)
        return null;
    return SerializableUtils.deserialize(sessionStrList.get(0));
}

@Override
protected void doUpdate(Session session) {
    if (session instanceof ValidatingSession
        && !((ValidatingSession) session).isValid()) {
        return;
    }
    String sql = "update sessions set session=? where id=?";
    jdbcTemplate.update(sql, SerializableUtils.serialize(session),
        session.getId());
}

@Override
protected void doDelete(Session session) {
    String sql = "delete from sessions where id=?";
    jdbcTemplate.update(sql, session.getId());
}
}

```

配置文件添加配置

```

<bean id="rememberMeCookie" class="org.apache.shiro.web.servlet.SimpleCookie">
    <constructor-arg value="rememberMe"/>
    <property name="httpOnly" value="true"/>
    <property name="maxAge" value="2592000"/> <!-- 30天-->
</bean>

<!-- rememberMe管理器-->
<bean id="rememberMeManager"
    class="org.apache.shiro.web.mgt.CookieRememberMeManager">
    <property name="cipherKey" value="
        #{T(org.apache.shiro.codec.Base64).decode('4AvVhmFLUs0KTA3KprsdaG==')}" />
    <property name="cookie" ref="rememberMeCookie"/>
</bean>

```

```
<!-- Session ID 生成器-->
<bean id="sessionIdGenerator"
      class="org.apache.shiro.session.mgt.eis.JavaUuidSessionIdGenerator"/>

<!-- Session DAO. 继承 EnterpriseCacheSessionDAO -->
<bean id="sessionDAO"
      class="com.atguigu.shiro.realms.MySessionDao">
  <property name="activeSessionsCacheName" value="shiro-activeSessionCache"/>
  <property name="sessionIdGenerator" ref="sessionIdGenerator"/>
</bean>

<!-- 会话管理器-->
<bean id="sessionManager" class="org.apache.shiro.web.session.mgt.DefaultWebSessionManager">
  <property name="globalSessionTimeout" value="1800000"/>
  <property name="deleteInvalidSessions" value="true"/>
  <property name="sessionValidationSchedulerEnabled" value="true"/>
  <property name="sessionDAO" ref="sessionDAO"/>
</bean>

<!-- SecurityManager -->
<bean id="securityManager" class="org.apache.shiro.web.mgt.DefaultWebSecurityManager"

      <property name="cacheManager" ref="cacheManager"/>
      <property name="rememberMeManager" ref="rememberMeManager"/>
      <property name="sessionManager" ref="sessionManager"> </property>
</bean>
```