

```
#include <iostream>
#include <iterator>
#include <vector>
#include <typeinfo>

using namespace std;

int main() {
    vector<string> vec = {"1", "2", "3"};

    using traits = iterator_traits<vector<string>::iterator>;

    cout << std::boolalpha;

    cout << "value_type == string : " << (typeid(traits::value_type) == typeid(string))
    << "\n";
    cout << "pointer == string* : " << (typeid(traits::pointer) == typeid(string*)) << "
    \n";
    cout << "reference == string& : " << (typeid(traits::reference) == typeid(string&))
    << "\n";
    cout << "difference_type == std::ptrdiff_t : " <<
        (typeid(traits::difference_type) == typeid(std::ptrdiff_t)) << "\n";

    cout << "iterator_category == std::random_access_iterator_tag : " <<
        (typeid(traits::iterator_category) == typeid(std::random_access_iterator_tag)) <<
        "\n";

    return 0;
}
```